```
                    OPERATING SYSTEM =  NOS 2.8.7 871/871.   23/09/20. PRINTED = 24/05/02. 11.20.34.

                    UJN          =  AAAI        FAMILY    = CYBER         JOB ORIGIN    = INTERACTIVE.
                    CREATING JSN =  AABA        USER NAME = INSTALL       SERVICE CLASS = INTERACTIVE.



          AAAAAAAAAA    AAAAAAAAAA    AAAAAAAAAA   IIIIIIIIIIII   AAAAAAAAAA    AAAAAAAAAA   BBBBBBBBBBB   BBBBBBBBBBB
         AAAAAAAAAAAA  AAAAAAAAAAAA  AAAAAAAAAAAA  IIIIIIIIIIII  AAAAAAAAAAAA  AAAAAAAAAAAA  BBBBBBBBBBBB  BBBBBBBBBBBB
         AA        AA  AA        AA  AA        AA       II       AA        AA  AA        AA  BB        BB  BB        BB
         AA        AA  AA        AA  AA        AA       II       AA        AA  AA        AA  BB        BB  BB        BB
         AA        AA  AA        AA  AA        AA       II       AA        AA  AA        AA  BB        BB  BB        BB
         AA        AA  AA        AA  AA        AA       II       AA        AA  AA        AA  BB        BB  BB        BB
         AA        AA  AA        AA  AA        AA       II       AA        AA  AA        AA  BB        BB  BB        BB
         AAAAAAAAAAAA  AAAAAAAAAAAA  AAAAAAAAAAAA       II       AAAAAAAAAAAA  AAAAAAAAAAAA  BBBBBBBBBBBB  BBBBBBBBBBBB
         AAAAAAAAAAAA  AAAAAAAAAAAA  AAAAAAAAAAAA       II       AAAAAAAAAAAA  AAAAAAAAAAAA  BBBBBBBBBBB   BBBBBBBBBBB
         AA        AA  AA        AA  AA        AA       II       AA        AA  AA        AA  BB        BB  BB        BB
         AA        AA  AA        AA  AA        AA       II       AA        AA  AA        AA  BB        BB  BB        BB
         AA        AA  AA        AA  AA        AA       II       AA        AA  AA        AA  BB        BB  BB        BB
         AA        AA  AA        AA  AA        AA       II       AA        AA  AA        AA  BB        BB  BB        BB
         AA        AA  AA        AA  AA        AA       II       AA        AA  AA        AA  BB        BB  BB        BB
         AA        AA  AA        AA  AA        AA  IIIIIIIIIIII  AA        AA  AA        AA  BBBBBBBBBBBB  BBBBBBBBBBBB
         AA        AA  AA        AA  AA        AA  IIIIIIIIIIII  AA        AA  AA        AA  BBBBBBBBBBB   BBBBBBBBBBB



        MWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWM
        WMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMW
        MWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWM
        WMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMW
        MWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWMWM
```

REC	NAME	TYPE	LENGTH	CKSUM	DATE	COMMENTS
1	CHECKA4	UPL*	1130507	0424		

CORRECTION IDENTS LISTED IN ORDER OF INSERTION.

COMDKS	CCOMISC	CCOMLOL	CGHCRMD	COMADEF	COMAREG	FA=CLO	FA=DEFS
FA=EOF	FA=EOR	FA=FLSH	FA=OPE	FA=RDC	FA=RDW	FA=RWX	FA=SET
FA=WTW	OPRDEFS	PSODEFS	SSTCALL	*TEXTS*	*CCG*	CGIA	MACROS
CGTM	*CT*	CFTBLD	IOTBLD	GPOCOM	UDTBLD	MIO	FBV
GPO	GRA	PROSEQ	SQZ	MCG	BDT	DMPIIT	DMPRLST
DMPSIT	DMPTREE	DMPUDI	PRNTGRA	PRNTMIO	PRNTUDI	PRNTABV	PRNTBV
PRNTRLI	PRNTMCG	CFA	UDT	FCC0391	FCC0392	FCC0393	FCC0408
FCC0414	FCC0415	FCC0448	FCC0502	FCC0504	FCC0508	FCC0509	FCC0513
FCC0534	FEAT121A	FCC0514	FCC0535	FCC0584	FCC0586	FCC0588	FCC0594
FCC0603	FCC0612	FCC0613	FCC0625	FCC0634	FCC0640	FCC0641	FCC0642
FCC0643	FCC0645	FCC0646	FCC0651	FCC0658	FCC0602	FCC0656	FCC0716
FCC0672	FCC0675	FCC0715	FCC0719	FCC0735	FCC0789	FCC0791	FCC0792
FCC0717	FCC0764	FCC0811	FCC0812	FCC0814	FCC0839	FCC0841	FCC0872
FCC0882	FCC0883	FCC0885	FCC0912	FCC0919	S3143FC	FCC0715A	FCC0777
FCC0881	FCC0909	FCC0914	FCC0977	FCC1014	FCC1025	FEAT155A	FEAT155AB
FCC0957	FCC0985	FEAT155F	S3143FCA	FCC0807	FCC1025A	FCC1029	FCC1039
FCC1040	FCC1056	FCC1085	FCC1088	FCC1107	FCC1123	FCC1126	FCC1127
FCC1191	FCC1196	FCC1227	FCC1239	FCC1254	FCC1290	FCCT001	FCC1319
FCC1322	FCC1352	FCC1353	FCC1413	FCC1424	FCC1442	FCC1451	F7540FT
FCC1491X	FCC1498X	FCC1530X	RSM2763X	FCCT005	FCC1504X	FCC1531X	FCC1532X
FCC1555X	FCC1556X	FCC1518X	RSM2763AX	FCC1568X	FCC1580X	CCG1	CCG2
CCG3	CCG4	CCG5	MCGF1	FCC1596X	FCC1623X	CCGDBG	FCC1626
FCC1644X	FCC1646X	SMOCDEF	CCG7	CCG8	CCG9	HISTORY	CCOMGCM
CCGDBGM	CCGILFD	FCC0885XA	FCC1655X	FCC1656X	FCC1660X	FCC1584X	FCC1677X
FCC1695X	FCC1700X	FCC1710X	FCC1719X	CCGNC	CCG11	CCG12	FCC1739X
FCC1740X	CMPLTXT	CCGTEXT	CCG13	CCG14	FCC1755X	FCC1770X	FCC1825X
FCC1828X	FCC1837X	FCC1842X	FCC1845X	FCC1858X	CCGTMTV	CCG15	CCG16
CCG17	FCC0238AX	CCG18	CCG19	CCG*78	FCC1908	FCC1918	FCC1928
CCGA002	CCGA005	CCG20	CCG21	CCG22	CCG23	CCGA006	CCGA007
CCG24	FCC1997	CCG25	CCG26	OUTPTK	CCG27	CCG28	FCC2038
FCC2045	FCC2056	FCC2051	CCG29	CCGA008	FCC2072	CCG30	CCG31
CCG33	COMAIDP	COMATOK	COMCBUB	COMCBUN	COMCIDP	COMCLFM	COMCMCS
COMCSBM	COMCTOK	COMDTOK	COMQSVR	COMSIDP	COMSTOK	CCOMRPV	DBG=MAC
DBG=IDP	CCG36	CCG32	CCG34	CCG35	FCC2074	FCC2091	FCCA462
CCG37	CCG38	CCG39	CCG40	FCC1040A	FCC2092	FCC2124	FCC2135
FCC2149	FCC2157	FCC2165	CCG41	*L498*	CCGA009	*L505*	F2950CC
CCGA010	CCGA011	CCGA012	CCGA013	CCGA014	CCGA015	*L508*	FCCA499X
FCCA506X	FCC2212X	CCGA017	CCGA018	CCGA019	CCGA021	CCGA022	CCGA024
L518	CCG*80	CCGA023	CCGA025	CCGA026	FCCA509X	FCCA513X	FCC2230X
FCC2271X	FCC2273X	*L528*	F1037CC	CCGA027	CCGA028	CCGA029	CCGA030
CCGA031	CCGA034	FCCA546X	FCCA550X	*L538*	CCG*81	CCGA035	CCGA036
CCGA037	CCGA039	*L552*	CCGA040	CCGA041	CCGA042	CCGA043	CCGA044
CCGA045	CCGA046	CCGA047	CCG0001	CCG0004	*L564*	CCG*82	CCGA049
CCGA052	CCGA054	CCG0006	CCG0013	*L577*	CCGA062	CCGA063	CCGA066
CCGA067	CCGA068	CCGA072	CCGA073	CCG0002	CCG0008	CCG0018	CCG0022
L587	CCGA076	FA=WTH	CCGA079	CCGA082	CCG0037	CCG0041	CCG0045
CCG0071	CCG0085	*L601*	CCGA083	CCG0012	CCG0034	CCG0057	CCG0086
CCG0088	CCG0089	*L617*	CCG0093	CCG0110	CCG0117	CCG0119	CCG0124
L628	CCGA085	CCGA088	CCG0127	CCG0128	CCG0146	CCG0147	CCG0150
CCG0154	*L642*	CCGA090	CCG0157	CCG0158	CCG0165	*L650*	CCG0172
CCG0181	*L670*	CCGA091	CCGA092	CCGA093	CCG0187	*L688*	CCGA094
CCG0177	CCG0188	CCG0189	CCG0191	CCG0193	CCG0196	*L716*	CCG0206
CCG0207	CCG0208	CCG0211	CCG0212	CCG0213	*L739*	CCG0214	CCG0215
CCG0216	CCG0218	CCG0220	*L780*	CCGA097	*L797*	CCGA099	CCG1*92

```
          ITEMIZE OF NEWPL          FILE     1              ** UNLABELED **              24/05/02. 11.17.17.      PAGE     2
     REC  NAME      TYPE          LENGTH   CKSUM     DATE   COMMENTS                                          ITEMIZE - 871

                     *L803*    CCGA100   *L826*    CCG*94    *L840*    *L847*    *L851*    *L859*
                     *L871*

          DECK LIST.
                     HISTORY   *COMDKS*  CCGTMTV   CCOMGCM   CCOMISC   CCOMLOL   CGHCRMD   COMADEF
                     COMAIDP   COMAREG   COMATOK   COMCBUB   COMCBUN   COMCIDP   COMCLFM   COMCMCS
                     COMCSBM   COMCTOK   COMDTOK   COMQSVR   COMSIDP   COMSTOK   CCOMRPV   DBG=MAC
                     DBG=IDP   FA=CLO    FA=DEFS   FA=EOF    FA=EOR    FA=FLSH   FA=OPE    FA=RDC
                     FA=RDW    FA=RWX    FA=SET    FA=WTH    FA=WTW    OPRDEFS   PSODEFS   CCGDBGM
                     CCGILFD   SMOCDEF   SSTCALL   *TEXTS*   CMPLTXT   WEOR1*    CCGTEXT   WEOR2*
                     *CCG*     CGIA      MACROS    CGTM      OUTPTK    *CT*      CFTBLD    IOTBLD
                     GPOCOM    UDTBLD    MIO       FBV       GPO       GRA       PROSEQ    SQZ
                     MCG       BDT       CCGDBG    DMPIIT    DMPRLST   DMPSIT    DMPTREE   DMPUDI
                     PRNTGRA   PRNTMIO   PRNTUDI   PRNTABV   PRNTBV    PRNTRLI   PRNTMCG   CFA
                     UDT

        2 * EOF *       SUM =    1130507

          ITEMIZE OF NEWPL          FILE     2              ** UNLABELED **              24/05/02. 11.17.17.      PAGE     2
     REC  NAME      TYPE          LENGTH   CKSUM     DATE   COMMENTS                                          ITEMIZE - 871

          * EOI *       SUM =    1130507
```

1412THE

1

1412THE

1

CORRECTION IDENTS ARE LISTED IN CHRONOLOGICAL ORDER OF INSERTION

COMDKS	CCOMISC	CCOMLOL	CGHCRMD	COMADEF	COMAREG	FA=CLO	FA=DEFS
FA=EOF	FA=EOR	FA=FLSH	FA=OPE	FA=RDC	FA=RDW	FA=RWX	FA=SET
FA=WTW	OPRDEFS	PSODEFS	SSTCALL	*TEXTS*	*CCG*	CGIA	MACROS
CGTM	*CT*	CFTBLD	IOTBLD	GPOCOM	UDTBLD	MIO	FBV
GPO	GRA	PROSEQ	SQZ	MCG	BDT	DMPIIT	DMPRLST
DMPSIT	DMPTREE	DMPUDI	PRNTGRA	PRNTMIO	PRNTUDI	PRNTABV	PRNTBV
PRNTRLI	PRNTMCG	CFA	UDT	FCC0391	FCC0392	FCC0393	FCC0408
FCC0414	FCC0415	FCC0448	FCC0502	FCC0504	FCC0508	FCC0509	FCC0513
FCC0534	FEAT121A	FCC0514	FCC0535	FCC0584	FCC0586	FCC0588	FCC0594
FCC0603	FCC0612	FCC0613	FCC0625	FCC0634	FCC0640	FCC0641	FCC0642
FCC0643	FCC0645	FCC0646	FCC0651	FCC0658	FCC0602	FCC0656	FCC0716
FCC0672	FCC0675	FCC0715	FCC0719	FCC0735	FCC0789	FCC0791	FCC0792
FCC0717	FCC0764	FCC0811	FCC0812	FCC0814	FCC0839	FCC0841	FCC0872
FCC0882	FCC0883	FCC0885	FCC0912	FCC0919	S3143FC	FCC0715A	FCC0777
FCC0881	FCC0909	FCC0914	FCC0977	FCC1014	FCC1025	FEAT155A	FEAT155AB
FCC0957	FCC0985	FEAT155F	S3143FCA	FCC0807	FCC1025A	FCC1029	FCC1039
FCC1040	FCC1056	FCC1085	FCC1088	FCC1107	FCC1123	FCC1126	FCC1127
FCC1191	FCC1196	FCC1227	FCC1239	FCC1254	FCC1290	FCCT001	FCC1319
FCC1322	FCC1352	FCC1353	FCC1413	FCC1424	FCC1442	FCC1451	F7540FT
FCC1491X	FCC1498X	FCC1530X	RSM2763X	FCCT005	FCC1504X	FCC1531X	FCC1532X
FCC1555X	FCC1556X	FCC1518X	RSM2763AX	FCC1568X	FCC1580X	CCG1	CCG2
CCG3	CCG4	CCG5	MCGF1	FCC1596X	FCC1623X	CCGDBG	FCC1626
FCC1644X	FCC1646X	SMOCDEF	CCG7	CCG8	CCG9	HISTORY	CCOMGCM
CCGDBGM	CCGILFD	FCC0885XA	FCC1655X	FCC1656X	FCC1660X	FCC1584X	FCC1677X
FCC1695X	FCC1700X	FCC1710X	FCC1719X	CCGNC	CCG11	CCG12	FCC1739X
FCC1740X	CMPLTXT	CCGTEXT	CCG13	CCG14	FCC1755X	FCC1770X	FCC1825X
FCC1828X	FCC1837X	FCC1842X	FCC1845X	FCC1858X	CCGTMTV	CCG15	CCG16
CCG17	FCC0238AX	CCG18	CCG19	CCG*78	FCC1908	FCC1918	FCC1928
CCGA002	CCGA005	CCG20	CCG21	CCG22	CCG23	CCGA006	CCGA007
CCG24	FCC1997	CCG25	CCG26	OUTPTK	CCG27	CCG28	FCC2038
FCC2045	FCC2056	FCC2051	CCG29	CCGA008	FCC2072	CCG30	CCG31
CCG33	COMAIDP	COMATOK	COMCBUB	COMCBUN	COMCIDP	COMCLFM	COMCMCS
COMCSBM	COMCTOK	COMDTOK	COMQSVR	COMSIDP	COMSTOK	CCOMRPV	DBG=MAC
DBG=IDP	CCG36	CCG32	CCG34	CCG35	FCC2074	FCC2091	FCCA462
CCG37	CCG38	CCG39	CCG40	FCC1040A	FCC2092	FCC2124	FCC2135
FCC2149	FCC2157	FCC2165	CCG41	*L498*	CCGA009	*L505*	F2950CC
CCGA010	CCGA011	CCGA012	CCGA013	CCGA014	CCGA015	*L508*	FCCA499X
FCCA506X	FCC2212X	CCGA017	CCGA018	CCGA019	CCGA021	CCGA022	CCGA024
L518	CCG*80	CCGA023	CCGA025	CCGA026	FCCA509X	FCCA513X	FCC2230X
FCC2271X	FCC2273X	*L528*	F1037CC	CCGA027	CCGA028	CCGA029	CCGA030
CCGA031	CCGA034	FCCA546X	FCCA550X	*L538*	CCG*81	CCGA035	CCGA036
CCGA037	CCGA039	*L552*	CCGA040	CCGA041	CCGA042	CCGA043	CCGA044
CCGA045	CCGA046	CCGA047	CCG0001	CCG0004	*L564*	CCG*82	CCGA049
CCGA052	CCGA054	CCG0006	CCG0013	*L577*	CCGA062	CCGA063	CCGA066
CCGA067	CCGA068	CCGA072	CCGA073	CCG0002	CCG0008	CCG0018	CCG0022
L587	CCGA076	FA=WTH	CCGA079	CCGA082	CCG0037	CCG0041	CCG0045
CCG0071	CCG0085	*L601*	CCGA083	CCG0012	CCG0034	CCG0057	CCG0086
CCG0088	CCG0089	*L617*	CCG0093	CCG0110	CCG0117	CCG0119	CCG0124
L628	CCGA085	CCGA088	CCG0127	CCG0128	CCG0146	CCG0147	CCG0150
CCG0154	*L642*	CCGA090	CCG0157	CCG0158	CCG0165	*L650*	CCG0172
CCG0181	*L670*	CCGA091	CCGA092	CCGA093	CCG0187	*L688*	CCGA094
CCG0177	CCG0188	CCG0189	CCG0191	CCG0193	CCG0196	*L716*	CCG0206
CCG0207	CCG0208	CCG0211	CCG0212	CCG0213	*L739*	CCG0214	CCG0215
CCG0216	CCG0218	CCG0220	*L780*	CCGA097	*L797*	CCGA099	CCG1*92

```
*L803*    CCGA100   *L826*    CCG*94    *L840*    *L847*    *L851*    *L859*
*L871*

        18 PURGED IDENTS WERE FOUND


DECKS ARE LISTED IN THE ORDER OF THEIR OCCURRENCE ON A NEW PROGRAM LIBRARY IF ONE IS CREATED BY THIS UPDATE

          YANK$$$   HISTORY   *COMDKS*  CCGTMTV   CCOMGCM   CCOMISC   CCOMLOL   CGHCRMD
          COMADEF   COMAIDP   COMAREG   COMATOK   COMCBUB   COMCBUN   COMCIDP   COMCLFM
          COMCMCS   COMCSBM   COMCTOK   COMDTOK   COMQSVR   COMSIDP   COMSTOK   CCOMRPV
          DBG=MAC   DBG=IDP   FA=CLO    FA=DEFS   FA=EOF    FA=EOR    FA=FLSH   FA=OPE
          FA=RDC    FA=RDW    FA=RWX    FA=SET    FA=WTH    FA=WTW    OPRDEFS   PSODEFS
          CCGDBGM   CCGILFD   SMOCDEF   SSTCALL   *TEXTS*   CMPLTXT   WEOR1*    CCGTEXT
          WEOR2*    *CCG*     CGIA      MACROS    CGTM      OUTPTK    *CT*      CFTBLD
          IOTBLD    GPOCOM    UDTBLD    MIO       FBV       GPO       GRA       PROSEQ
          SQZ       MCG       BDT       CCGDBG    DMPIIT    DMPRLST   DMPSIT    DMPTREE
          DMPUDI    PRNTGRA   PRNTMIO   PRNTUDI   PRNTABV   PRNTBV    PRNTRLI   PRNTMCG
          CFA       UDT

 THIS UPDATE REQUIRED  62200B WORDS OF MEMORY.
```

SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - YANK$$$

```
          IDENTIFIER                TOTAL    ACTIVE


          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN HISTORY

HISTORY  *COMDECK  HISTORY                                                  HISTORY        1     A
HISTORY   FCC0885XA GPO/MII - CHANGE ILD TO TLD IF IH = IT. SO CODE IS CORRECT  FCC0885XA      1     A
HISTORY            WHEN WE MOVE AN INVARIANT FUNCTION REFERENCE FROM AN INNER LP.  FCC0885XA      2     A
HISTORY            SIJ    4/7/77      APPLIES TO FTN 4.6, OPT=2             FCC0885XA      3     A
HISTORY                                                                     FCC0885XA      4     A
HISTORY   FCC1655X MCG - RESOLVE DEADLOCK SITUATIONS IN A WIDER VARIETY OF CASES  FCC1655X       1     A
HISTORY            SIJ    77/02/28    APPLIES TO FTN 4.6                    FCC1655X       2     A
HISTORY                                                                     FCC1655X       3     A
HISTORY   FCC1656X MCG/JAM/CRC - COUNT A INTERFERING STORE AS A USE WHEN    FCC1656X       1     A
HISTORY            CALCULATING THE NEXT USE DISTANCE.                       FCC1656X       2     A
HISTORY            SIJ    2/9/77      APPLIES TO FTN 4.6, OPT=1,2           FCC1656X       3     A
HISTORY                                                                     FCC1656X       4     A
HISTORY   FCC1660X SQZ/IA&IS - DO NOT REDUCE (I+-J)-+I COMBINATIONS IF FIRST IS  FCC1660X       1     A
HISTORY            AN INCREMENT AND THEY SPAN BASIC BLOCKS.                 FCC1660X       2     A
HISTORY                                                                     FCC1660X       3     A
HISTORY            SIJ    2/24/77     APPLIES TO FTN 4.6, OPT=1,2           FCC1660X       4     A
HISTORY                                                                     FCC1660X       5     A
HISTORY   FCC1584X GPO/MII/LD - DO NOT MOVE INDEXED LD OFF A SIDE BRANCH IF THE  FCC1584X       1     A
HISTORY            SUBSCRIPT IS NOT A LD OF A SIMPLE INTEGER VARIABLE.      FCC1584X       2     A
HISTORY            SIJ    3/9/77      APPLIES TO FTN 4.6, OPT=1,2           FCC1584X       3     A
HISTORY                                                                     FCC1584X       4     A
HISTORY   FCC1677X PROSEQ/ESR - TRANSFER UDT ORDINAL FROM LD/ST TO STT FOR USE BY  FCC1677X       1     A
HISTORY            GRA SO WE DONT KILL STORES OF SIMPLE LCM VARIABLES .     FCC1677X       2     A
HISTORY            SIJ    3/15/77     APPLIES TO FTN 4.6, OPT=2             FCC1677X       3     A
HISTORY                                                                     FCC1677X       4     A
HISTORY   FCC1695X BDT/FIL - FORM INTERFERENCE LINKS BETWEEN CONDITIONAL JUMPS  FCC1695X       1     A
HISTORY            WHEN CALLED FROM *RIO* SO IT REORDERS THE SEQUENCE CORRECTLY.  FCC1695X       2     A
HISTORY            SIJ    3/16/77     APPLIES TO FTN 4.6, OPT=1,2           FCC1695X       3     A
HISTORY                                                                     FCC1695X       4     A
HISTORY   FCC1700X PURGE FCC0715 SINCE IT IS NO LONGER NECESSARY.           FCC1700X       1     A
HISTORY            SIJ    3/28/77     APPLIES TO FTN 4.6, OPT=2             FCC1700X       2     A
HISTORY                                                                     FCC1700X       3     A
HISTORY   FCC1710X GPO/CIF - WHEN AN *IP* EXPRESSION IS OF THE FORM (I+INVEXP)-I,  FCC1710X       1     A
HISTORY            ETC IS REDUCED BY *SQZ* TO *INVEXP* DO NOT FORM THE *IP*  FCC1710X       2     A
HISTORY            FORMULAS IN *IIT* FOR IT.                                FCC1710X       3     A
HISTORY            SIJ    3/31/77     APPLIES TO FTN 4.6, OPT=1,2           FCC1710X       4     A
HISTORY                                                                     FCC1710X       5     A
HISTORY   FCC1719X SQZ/IS - DO NOT CALL *CIA* IF HAVE  IS I,CON,K SO WE DONT  FCC1719X       1     A
HISTORY            REDUCE K = CON-K TO K = K-CON AFTER A LOOP.              FCC1719X       2     A
HISTORY            SIJ    4/1/77      APPLIES TO FTN 4.6, OPT=2             FCC1719X       3     A
HISTORY                                                                     FCC1719X       4     A
HISTORY   FCC1739X BDT - CLEAR *TSTC* AFTER *FIL* SO IT IS RESET WHEN *GAS* IS  FCC1739X       1     A
HISTORY            CALLED TO GET MORE SPACE.                                FCC1739X       2     A
HISTORY            SIJ    5/1/77      APPLIES TO FTN 4.6, OPT               FCC1739X       3     A
HISTORY                                                                     FCC1739X       4     A
HISTORY   FCC1740X GRA/IRA/XMT - IF PRED HAS NO X-VALUE THEN USE B-REG VALUE  FCC1740X       1     A
HISTORY            AND GENERATE A *SA* INSTEAD.                             FCC1740X       2     A
HISTORY            SIJ    5/1/77      APPLIES TO FTN 4.6, OPT=2             FCC1740X       3     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN HISTORY

```
HISTORY                                                                  FCC1740X     4     A
HISTORY   FCC1755X UDT/PBB - SET  THE FINAL JUMP BIT IN THE *BIT* ENTRY OF  FCC1755X     1     A
HISTORY            A BLOCK THAT HAS A *JIN* INSTRUCTION.                 FCC1755X     2     A
HISTORY            GJM    06/01/77    APPLIES TO FTN 4.6,OPT = 2         FCC1755X     3     A
HISTORY                                                                  FCC1755X     4     A
HISTORY   FCC1770X GRA/ERC/RS - DO NOT SET *IRA* BIT FOR CASE UL = N $ DO I =1,UL  FCC1770X     1     A
HISTORY            WHERE N IS A F.P. AND IS CHANGED TO A *AC* IN THE OUTER LOOP  FCC1770X     2     A
HISTORY            BECAUSE OF REFS IN THE INNER LOOP.                    FCC1770X     3     A
HISTORY            SIJ    5/26/77     APPLIES TO FTN 4.6, OPT=2          FCC1770X     4     A
HISTORY                                                                  FCC1770X     5     A
HISTORY   FCC1825X BDT/RIO - FORCE LATE OUTPUT OF INSTRUCTIONS THAT *PRS* TO AN  FCC1825X     1     A
HISTORY            X-REGISTERS IF THEY HAVE NO OPERAND PREDECESSORS.     FCC1825X     2     A
HISTORY            SIJ    8/9/77      APPLIES TO FTN 4.6, OPT=2          FCC1825X     3     A
HISTORY                                                                  FCC1825X     4     A
HISTORY   FCC1828X GRA/MFA - DO NOT ADD ADDRESS CANIDATES WITH A H2 TO THE MFA  FCC1828X     1     A
HISTORY            ADDRESS CANIDATE TABLE SINCE THERE IS NOT ROOM FOR THE H2  FCC1828X     2     A
HISTORY            FIELD IN IT.                                          FCC1828X     3     A
HISTORY            SIJ    77/08/04    APPLIES TO FTN 4.6, OPT=2          FCC1828X     4     A
HISTORY                                                                  FCC1828X     5     A
HISTORY   FCC1837X PROSEQ/AIS - SET LAST USE ADDRESS OF INSTRUCTION THAT *PRS*  FCC1837X     1     A
HISTORY            A UJP OR FULL LOCK *RS* TO END OF SUBSEQUENCE SO WE DONT  FCC1837X     2     A
HISTORY            END UP WITH A B-REGISTER DEADLOCK IN *MCG*.           FCC1837X     3     A
HISTORY            SIJ    8/24/77     APPLIES TO FTN 4.6, OPT=2          FCC1837X     4     A
HISTORY                                                                  FCC1837X     5     A
HISTORY   FCC1845X MCG/CRA - ALLOW OVERRIDE OF FULL LOCK RS"S TO B-REGISTERS  FCC1845X     1     A
HISTORY            SO A SWAP(BI,BJ) SEQUENCE OF INSTRUCTIONS WORKS.      FCC1845X     2     A
HISTORY            SIJ    8/29/77     APPLIES TO FTN 4.6, OPT=1,2        FCC1845X     3     A
HISTORY                                                                  FCC1845X     4     A
HISTORY   FCC1858X GPO - ENSURE 0 TERMINATOR AT END OF *BIT* WHEN INITIALIZING IT  FCC1858X     1     A
HISTORY            SIJ    9/21/77     APPLIES TO FTN 4.6, OPT=2          FCC1858X     2     A
HISTORY                                                                  FCC1858X     3     A
HISTORY   FCC0238AX GPO/MII - MARK INVARIANT *IM* AS UNSAFE IF ANY OF ITS  FCC0238AX     1     A
HISTORY            OPERANDS MAY CAUSE AN INTERRUPT WHEN THE *IM* IS EXECUTED.  FCC0238AX     2     A
HISTORY            SQZ - INSERT AN UNPACK INSTRUCTION PRIOR TO UNSAFE IM.  FCC0238AX     3     A
HISTORY            SIJ    6/6/77      APPLIES TO FTN 4.6, OPT=1,2        FCC0238AX     4     A
HISTORY                                                                  FCC0238AX     5     A
HISTORY   CCGA004  ADD COPYRIGHT INFORMATION TO OLDPL                    CCG*78       1     A
HISTORY            SIJ    2/2/78                                         CCG*78       2     A
HISTORY                                                                  CCG*78       3     A
HISTORY   FCC1908  BDT/SIO - DO NOT STACK STORES IF RF > RI AND RF NOT PROCESSED  FCC1908     1     A
HISTORY            WHEN THE RI IS ENCOUNTERED.                           FCC1908     2     A
HISTORY            SIJ    3/8/78      APPLIES TO FTN 4.6                 FCC1908     3     A
HISTORY                                                                  FCC1908     4     A
HISTORY   FCC1918  GRA/OSI - MARK STT"S THAT SETUP INITIAL ADDRESS"S WITH THE  FCC1918     1     A
HISTORY            *SIA* BIT SO THAT PROSEQ/ESR KNOWS TO TRUNCATE THE ADDRESS  FCC1918     2     A
HISTORY            (LCM AND FP) TO 18 BITS TO AVOID PROBLEMS WHEN THE SUBSCRIPT  FCC1918     3     A
HISTORY            EXPRESSION IS NEGATIVE.                               FCC1918     4     A
HISTORY            SIJ    2/2/78      APPLIES TO FTN 4.6, OPT=2 ON 7600 ONLY  FCC1918     5     A
HISTORY                                                                  FCC1918     6     A
HISTORY   FCC1928  SQZ/SA - DONT"T SET RF BIT FOR PRED SO WE DONT INCORRECTLY  FCC1928     1     A
HISTORY            REDUCE 60 BIT ARITHMETIC TO 18 BITS IN GPO/GRA.       FCC1928     2     A
HISTORY            SQZ/STT - SET RF BIT ONLY IF IH " 0 FOR SAME REASONS.  FCC1928     3     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN HISTORY

```
HISTORY            SIJ    3/1/78      APPLIES TO FTN 4.6, OPT=2                    FCC1928    4    A
HISTORY                                                                            FCC1928    5    A
HISTORY   CCGA005  CCGTEXT - ADD A CONDITIONAL LOAD FACILITY TO CGIA                CCGA005    1    A
HISTORY            VIA A SECOND PARAMETER IN THE *USE* PSEUDO                       CCGA005    2    A
HISTORY            SIJ    2/23/78                                                   CCGA005    3    A
HISTORY                                                                            CCGA005    4    A
HISTORY   CCGA006  MCG - FIX MISC BUGS IN JAM MODE THAT ONLY SHOWED UP WHEN         CCGA006    1    A
HISTORY            PL/I WAS THE HOST..  CORRECTLY ZERO THE USES COUNT FOR           CCGA006    2    A
HISTORY            OMMITTED SXT"S SO WE DON"T ERRONEOUSLY STORE OUT REGISTERS.      CCGA006    3    A
HISTORY            CORRECTLY RESET N.SP AFTER AN *EOS*, *UJP*, ETC TO AVOID         CCGA006    4    A
HISTORY            SPURRIOUS CALLS TO JAM MODE.                                     CCGA006    5    A
HISTORY            DO NOT CALL *RIO* WITH FLAG=1 SINCE *BDT* MAY FOUL UP IF A       CCGA006    6    A
HISTORY            STORE STACK OCCURS.                                              CCGA006    7    A
HISTORY            SIJ    4/18/78     APPLIES TO CCG                                CCGA006    8    A
HISTORY                                                                            CCGA006    9    A
HISTORY   CCGA007  CGIA - SET CONDITIONAL LOAD BIT FOR REPI TABLES                  CCGA007    1    A
HISTORY            SIJ    6/6/78                                                    CCGA007    2    A
HISTORY                                                                            CCGA007    3    A
HISTORY   FCC1997  GRA/DUM - SET *EU* FOR INCREMENT CANIDATE IF STORE IS NOT NEXT   FCC1997    1    A
HISTORY            TO INCREMENT VARIABLE.                                           FCC1997    2    A
HISTORY            SIJ    6/6/78      APPLIES TO FTN 4.6, OPT=1,2                   FCC1997    3    A
HISTORY                                                                            FCC1997    4    A
HISTORY   CCG25    GRA - SPECIAL CASE LOOPS OF THE FORM -                           CCG25      1    A
HISTORY            DO LAB J =                                                       CCG25      2    A
HISTORY            A(I) = A(I) OP EXPR                                              CCG25      3    A
HISTORY            SO AS TO ASSIGN A(I) TO AN X-REGISTER IN THE LOOP                CCG25      4    A
HISTORY                                                                            CCG25      5    A
HISTORY   CCG26    CFA/MRB - IMPLEMENT LISTING OF DEAD CODE MESSAGES ACCORDING      CCG26      1    A
HISTORY            TO REVISED INTERFACE SPEC SECTION 3.2.69  CALL HOST ROUTINE      CCG26      2    A
HISTORY            *HR$LDC$ WITH TABLE OF LINE NUMBERS IN *RND*.                    CCG26      3    A
HISTORY            CCGTEXT - FIX LXQ, RPVDEF AND RPVFWA MACROS SO AS TO AGREE       CCG26      4    A
HISTORY            WITH FTN 5 MACRO DEFINITIONS9                                    CCG26      5    A
HISTORY                                                                            CCG26      6    A
HISTORY   CCG27    SQZ/JPX - ELIMINATE JPX"S IF OPERAND IS CONSTANT                 CCG27      1    A
HISTORY            SIJ    7/20/78                                                   CCG27      2    A
HISTORY                                                                            CCG27      3    A
HISTORY   FCC2038  MCG/CSI - AWWAYS START AT X6 AFTER ISSUING A STORE SINCE ISI     FCC2038    1    A
HISTORY            WILL NOT RETURN B3 SET PROPERLY IF A DWL WAS ISSUED.             FCC2038    2    A
HISTORY            SIJ    8/8/78      APPLIES TO FTN 4.6, OPT=1,2                   FCC2038    3    A
HISTORY                                                                            FCC2038    4    A
HISTORY   FCC2045  GPO/UPB - GET CORRECT TLD FOR XMT IN OPT=1 WHEN PROCESSING       FCC2045    1    A
HISTORY            ABNORMAL IP IN LOOP WITH MULTIPLE BLOCKS.                        FCC2045    2    A
HISTORY                                                                            FCC2045    3    A
HISTORY   FCC2056  GRA/DUM - MARK PRED OF RS AS *EU* SO WE DONT BIAS THE            FCC2056    1    A
HISTORY            ASSIGNMENT OF AN OUTER LOOP INDEX WHICH HAS BEEN ASSIGNED TO     FCC2056    2    A
HISTORY            A REGISTER IN A INNER LOOP.                                      FCC2056    3    A
HISTORY            SIJ    8/24/78     APPLIES TO FTN 4.7, OPT=2                     FCC2056    4    A
HISTORY                                                                            FCC2056    5    A
HISTORY   FCC2051  GRA/IRA - PROCESS HOLDING BLOCK OF INNER LOOP SO THAT ANY        FCC2051    1    A
HISTORY            EPILOGUE POST STORES THAT HAVE TO BE PLACED IN IT ARE.           FCC2051    2    A
HISTORY            SIJ    8/14/78     APPLIES TO FTN 4.7, OPT=2                     FCC2051    3    A
HISTORY                                                                            FCC2051    4    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN HISTORY

```
HISTORY   CCGA008  CGIA - FIX RELOCATION AFTER SET *                          CCGA008    1   A
HISTORY            SIJ/KIT  9/26/78                                           CCGA008    2   A
HISTORY                                                                       CCGA008    3   A
HISTORY   FCC2072  SQZ/DRL - IF LVL2 = DIRECT AND PREDECESSOR IS AN *STT* THEN FCC2072    1   A
HISTORY            REPLACE SEMANTIC INFO IN DRL/DWL WITH SEMANTIC INFO IN *STT* FCC2072    2   A
HISTORY            IN CASE *ARF* HAS CHANGED THE RF BECAUSE IT WAS *IAC* SO WE FCC2072    3   A
HISTORY            DON"T MODE OUT LATER ON.                                   FCC2072    4   A
HISTORY            SIJ    9/14/78     APPLIES TO FTN 4.7, OPT                 FCC2072    5   A
HISTORY                                                                       FCC2072    6   A
HISTORY   CCG31    FIX INITIALIZATION FOR CODE GENERATED IN LOCAL BLOCKS      CCG31      1   A
HISTORY   CCG36    GPO/MII,FII - EXTEND GPO TO HANDLE INCREMENTS WITH *LDC*"S 000120 CCG36      1   A
HISTORY            AND LARGE MASKS.                                           000130 CCG36      2   A
HISTORY            SIJ    11/15/78    APPLIES TO FTN 5                        000140 CCG36      3   A
HISTORY                                                                       000150 CCG36      4   A
HISTORY   CCG32    BDT/CRW - FIX PROCESSING OF TYPE IV INSTRUCTIONS  SO THAT 000540 CCG32      1   A
HISTORY            X-REGISTER ASSIGNMENT WORKS IN OPT=1 WHEN WE HAVE A MULTI 000550 CCG32      2   A
HISTORY            BLOCK LOOP.                                                000560 CCG32      3   A
HISTORY            SIJ    11/11/78                                            000570 CCG32      4   A
HISTORY                                                                       000580 CCG32      5   A
HISTORY   CCG34    SQZ,GRA - DO NOT SET RF BIT FOR STT IF IH = TRACE. SO      000710 CCG34      1   A
HISTORY            ASSIGNED GOTO PROCESSING WORKS CORRECTLY                   000720 CCG34      2   A
HISTORY            SIJ    11/16/78    APPLIES TO FTN 5                        000730 CCG34      3   A
HISTORY                                                                       000740 CCG34      4   A
HISTORY   CCG35    MISC LEVEL 0 CODE FIXS                                     001130 CCG35      1   A
HISTORY            SIJ AND MAH  11/20/78    APPLIES TO FTN 5                  001140 CCG35      2   A
HISTORY                                                                       001150 CCG35      3   A
HISTORY   FCC2074  GRA/ERC - DO NOT ENTER AN STT THAT IS *IRA* IF IH " 0 AND 002120 FCC2074      1   A
HISTORY            RF IS NOT A VALUE CANIDATE AS THIS IS AN ERROR  ( LOCF(A(FP)))002130 FCC2074      2   A
HISTORY            SIJ    9/15/78     APPLIES TO FTN 4.7, OPT=2               002140 FCC2074      3   A
HISTORY                                                                       002150 FCC2074      4   A
HISTORY   FCC2091  GPO/TRP - ALWAYS MAKE *HB* OF *FTH* LOOP REMOTE ( *UJP* AT 002310 FCC2091      1   A
HISTORY            END ) SINCE *TRP* DOESN"T KNOW IF POST STORES WILL BE      002320 FCC2091      2   A
HISTORY            INSERTED BY *IPS* LATER.                                   002330 FCC2091      3   A
HISTORY            SIJ    11/21/78    APPLIES TO FTN 4.6, OPT=2               002340 FCC2091      4   A
HISTORY                                                                       002350 FCC2091      5   A
HISTORY   FCCA462  GPO/MII - CHANGE INSTANCES OF 18 BIT ARITHMETIC IN JUMP    002420 FCCA462      1   A
HISTORY            PROCESSORS TO 60 BIT SO TOP BITS OF *N.LJ* DON"T GET DESTROYED002430 FCCA462      2   A
HISTORY            SIJ    11/24/78    APPLIES TO FTN 4.7, OPT=2               002440 FCCA462      3   A
HISTORY                                                                       002450 FCCA462      4   A
HISTORY   -I2B-    CY=    OF P21MODS AND CY=    OF CGMMODS PICKED UP          CCG38      1   A
HISTORY            FOR SYSTEM MODE BUILD.                                     CCG38      2   A
HISTORY            CGM    78-11-12    APPLIES TO FTN5PL, CY=50.               CCG38      3   A
HISTORY                                                                       CCG38      4   A
HISTORY   -I2D-    CY=    OF P22MODS AND CY=    OF CGMMODS PICKED UP          CCG39      1   A
HISTORY            FOR SYSTEM MODE BUILD.                                     CCG39      2   A
HISTORY            CGM    78-11-12    APPLIES TO FTN5PL, CY=46.               CCG39      3   A
HISTORY                                                                       CCG39      4   A
HISTORY   -R2A-    CY=    OF P23MODS AND CY=    OF CGMMODS PICKED UP          CCG40      1   A
HISTORY            FOR SYSTEM MODE BUILD.                                     CCG40      2   A
HISTORY            CGM    79-03-16    APPLIES TO FTN5PL, CY=52.               CCG40      3   A
HISTORY                                                                       CCG40      4   A
HISTORY   *L498*    ***********************************************          *L498*     1   A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN HISTORY

```
HISTORY   CCGA009  GRA. AVOID USE OF NON-INTERFACE SYMBOL *OPT2* TO PREVENT      CCGA009    1    A
HISTORY            ERROR IN PL1 BUILD.                                          CCGA009    2    A
HISTORY            MAH    79/5/07     APPLIES TO CCG.                           CCGA009    3    A
HISTORY                                                                         CCGA009    4    A
HISTORY   *L505*    ************************************************           *L505*     1    A
HISTORY   F2950CC  MIO/WMB,PROSEQ/AIS.  MODIFY CODE TO USE HF MICROS AND FIX    F2950CC    1    A
HISTORY            BAD PLUG CODE.                                               F2950CC    2    A
HISTORY            SPS    79/5/07     APPLIES TO CCG.                           F2950CC    3    A
HISTORY                                                                         F2950CC    4    A
HISTORY   CCGA010  BDT/FIL.  APPLY FCC0715 TO BDT.                              CCGA010    1    A
HISTORY            SPS    05-15-79    APPLIES TO CCG.                           CCGA010    2    A
HISTORY                                                                         CCGA010    3    A
HISTORY   CCGA011  CGTM/ATS,CCGTMTV.  IMPROVE DYNAMIC TABLE ALLOCATION SCHEME.  CCGA011    1    A
HISTORY            SPS    05-16-79    APPLIES TO CCG.                           CCGA011    2    A
HISTORY                                                                         CCGA011    3    A
HISTORY   CCGA012  MIO,UDT,OUTPTK/SETST.  ALLOW USE OF MAX AVAIL ECS. FIX SNAPS. CCGA012   1    A
HISTORY            SPS    05-14-79    APPLIES TO CCG.                           CCGA012    2    A
HISTORY                                                                         CCGA012    3    A
HISTORY   CCGA013  SQZ/BTERM.  CHECK CC$OPF FLAG FOR CONDITION TRUE JUMPS       CCGA013    1    A
HISTORY            BEFORE DEAD CODE IS MARKED IN CFT.                           CCGA013    2    A
HISTORY            SPS    05-16-79    APPLIES TO CCG.                           CCGA013    3    A
HISTORY                                                                         CCGA013    4    A
HISTORY   CCGA014  PROSEQ. FIX LEVEL 0 (CLOBBERED REGISTER).                    CCGA014    1    A
HISTORY   MAH      79/06/11            APPLIES TO FTN5 5.0                      CCGA014    2    A
HISTORY                                                                         CCGA014    3    A
HISTORY   CCGA015  COMATOK. FIX HOLLERITH IN 63-CHAR SET.                       CCGA015    1    A
HISTORY   MAH      79/06/11            APPLIES TO CCG .                         CCGA015    2    A
HISTORY                                                                         CCGA015    3    A
HISTORY   *L508*    ************************************************           *L508*     1    A
HISTORY   FCCA499X GPO/RDD.  RDD WAS ELIMINATING STORES WITHOUT SCANNING FOR    FCCA499X  1    A
HISTORY            INTERFERING USES (EQUIVALENCE CLASSES) AFTER THE STORE.      FCCA499X  2    A
HISTORY            SPS    8-9-79      APPLIES TO FCC,CCG                        FCCA499X  3    A
HISTORY                                                                         FCCA499X  4    A
HISTORY   FCCA506X  GPO/MII. AVOID MODE 1 IN GPO FOR TLD ENCOUNTERED.           FCCA506X  1    A
HISTORY   MAH      07/25/79            APPLIES TO FTN 4.8 AND CCG .             FCCA506X  2    A
HISTORY                                                                         FCCA506X  3    A
HISTORY   FCC2212X BDT/FTL.  FTL WAS NOT FORMING DEPENDENCY LINKS FROM A        FCC2212X  1    A
HISTORY            TEMPORARY *RS* TO FULL *RS*. IN ADDITION RIO WOULD OFTEN FLOAT FCC2212X  2    A
HISTORY            ALL FULL *RS* POSSIBLE IN A CLASS OF REGISTERS ABOVE A TEMP OF FCC2212X  3    A
HISTORY            THE SAME CLASS.  THIS MADE LIFE FOR MCG PRETTY TOUGH.        FCC2212X  4    A
HISTORY            SPS    9-13-79     APPLIES TO FTN4.8,CCG                     FCC2212X  5    A
HISTORY                                                                         FCC2212X  6    A
HISTORY   CCGA017  DBG=MAC,BDT.  TREE SNAPS IN BDT COULD NOT BE OBTAINED        CCGA017    1    A
HISTORY            BECAUSE SNAP MACROS WERE ACCESSING THE WRONG REGISTER        CCGA017    2    A
HISTORY            HOLDING LOCATIONS.                                           CCGA017    3    A
HISTORY            SPS    8-1-79      APPLIES TO CCG.                           CCGA017    4    A
HISTORY                                                                         CCGA017    5    A
HISTORY   CCGA018  CGTM,CCGTEXT,COMCIDP,FA=DEFS,FA=RDC,FA=RDW.                  CCGA018    1    A
HISTORY            SCOPE 2 SUPPORT CODE.                                        CCGA018    2    A
HISTORY            MAP    8-7-79      APPLIES TO CCG.                           CCGA018    3    A
HISTORY                                                                         CCGA018    4    A
HISTORY   CCGA019  MCG/UJR.  MCG MODES OUT WHEN THE X-REGISTER WIDTH IS         CCGA019    1    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN HISTORY

```
HISTORY            GREATER THAN 8 AND X6 AND X7 ARE LOCKED UP AS A RESULT OF          CCGA019     2    A
HISTORY            A PREVIOUS ALLOCATION.  THIS CODE BREAKS THE DEADLOCK BY           CCGA019     3    A
HISTORY            STORING OUT X6 OR X7 REGARDLESS OF ANY PREVIOUS COMMITMENTS.       CCGA019     4    A
HISTORY            X6 OR X7 IS THEN RESTORED WITH THE PROPER VALUE BEFORE             CCGA019     5    A
HISTORY            EXITING THE BLOCK WITHOUT DESTROYING THE CONTENTS OF THE           CCGA019     6    A
HISTORY            THE OTHER X REGISTERS.                                             CCGA019     7    A
HISTORY            SPS    8-02-79     APPLIES TO CCG                                  CCGA019     8    A
HISTORY                                                                               CCGA019     9    A
HISTORY   CCGA021  GPO,CGTM,CFA.  LABEL DEFINITION ELIMINATED BECAUSE                 CCGA021     1    A
HISTORY            THE *HB* OF A *FTH* LOOP COMBINED WITH ITS NATURAL PRED,           CCGA021     2    A
HISTORY            BUT REFS TO THE LABEL (ASSIGNED GOTOS) REMAIN.  ADD CODE           CCGA021     3    A
HISTORY            TO OUTPUT THE LABEL DEFINITION WITH THAT OF THE GENERATED          CCGA021     4    A
HISTORY            LABEL.                                                             CCGA021     5    A
HISTORY            SPS    8-16-79     APPLIES TO CCG.                                 CCGA021     6    A
HISTORY                                                                               CCGA021     7    A
HISTORY   CCGA022  COMATOK.  AVOID ASSEMBLY ERRORS WHEN USING COMATOK WITH            CCGA022     1    A
HISTORY            LEVEL 507 COMPASS.                                                 CCGA022     2    A
HISTORY            MAH    8-29-79     APPLIES TO CCG.                                 CCGA022     3    A
HISTORY                                                                               CCGA022     4    A
HISTORY   CCGA024  SQZ/ECO.  AVOID OVERFLOW INTERRUPTS ON 7600 WHILE SUBSUMING        CCGA024     1    A
HISTORY            FLOATING POINT OPERATIONS.                                         CCGA024     2    A
HISTORY            MAP    10-15-79    APPLIES TO CCG.                                 CCGA024     3    A
HISTORY                                                                               CCGA024     4    A
HISTORY   *L518*    ***********************************************                  *L518*      1    A
HISTORY   CCG*80   UPDATE COPYRIGHT INFORMATION FOR 1980                              CCG*80      1    A
HISTORY            BEA    79/09/28                                                    CCG*80      2    A
HISTORY                                                                               CCG*80      3    A
HISTORY   CCGA023  MCG/UJR  UJR IN SEEKING TO UNJAM REGISTERS WAS STORING OUT         CCGA023     1    A
HISTORY            AN OPERAND OF THE INSTRUCTION IT IS ATTEMPTING TO ISSUE.           CCGA023     2    A
HISTORY            THIS WOULD REMOVE THAT INSTRUCTION OF THE ISSUE CANDIDATE LIST     CCGA023     3    A
HISTORY            MAKING IT IMPOSSIBLE TO ISSUE BEFORE THE DESTROYED OPERAND         CCGA023     4    A
HISTORY            COULD BE RELOADED.  A MODULE WAS ADDED TO RECOGNIZE THIS           CCGA023     5    A
HISTORY            AND TOGGLE THE OPERAND REGISTER WITH A REGISTER OF THE NEEDED      CCGA023     6    A
HISTORY            CLASS WITHOUT A STORE TO MEMORY.  THIS WAS DONE BY ISSUEING A      CCGA023     7    A
HISTORY            SERIES OF XOR'S.   THIS IS CCG VERSION OF FCC2207.                 CCGA023     8    A
HISTORY            SPS    10-08-79     APPLIES TO CCG.                                CCGA023     9    A
HISTORY                                                                               CCGA023    10    A
HISTORY   CCGA025  UDT,CGTM.  MAKE MORE EFFICIENT USE OF FINAL OBTAINABLE             CCGA025     1    A
HISTORY            FIELD LENGTH.                                                      CCGA025     2    A
HISTORY            MAH    11-03-79    APPLIES TO CCG.                                 CCGA025     3    A
HISTORY                                                                               CCGA025     4    A
HISTORY   CCGA026  UDT/AUT.  BIT VECTOR POSITIONS WERE CONFUSED.  THIS CAUSED         CCGA026     1    A
HISTORY            GARBAGE IN DEF AND LX VECTOR OF THE ENTRY AND EXIT BLOCK. THIS     CCGA026     2    A
HISTORY            AFFECTED THE CALCULATION OF LX VECTORS FOR ALL BLOCKS.             CCGA026     3    A
HISTORY            SPS    10-5-79     APPLIES TO FTN.8,CCG.                           CCGA026     4    A
HISTORY                                                                               CCGA026     5    A
HISTORY   FCCA509X GPO/CBB,GRA/CLB,MIO.  CBB IN BUILDING AN *EBB* ASSUMMED            FCCA509X    1    A
HISTORY            THAT THE PHYSICAL PRED. OF A HB IN THE BST TABLE WAS               FCCA509X    2    A
HISTORY            ALWAYS THE LOGICAL PRED. AS WELL.  THIS IS NOT ALWAYS              FCCA509X    3    A
HISTORY            THE CASE AND ADDITIONAL CODE WAS ADDED TO CHECK FOR THIS.          FCCA509X    4    A
HISTORY            SPS    10-2-79     APPLIES TO FTN4.8,CCG.                          FCCA509X    5    A
HISTORY                                                                               FCCA509X    6    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN HISTORY

```
HISTORY   FCCA513X GPO/MII,FXI.  MII WAS MAKING INSUFFICIENT TEST ON          FCCA513X    1    A
HISTORY            WHETHER INVARIANT STORES COULD BE MOVED.  WHAT WAS         FCCA513X    2    A
HISTORY            NEEDED WAS A CHECK ON THE REGION LIVE ENTRY VECTOR (LEA).  FCCA513X    3    A
HISTORY            CODE WAS ADDED TO CALCULATE LEA BEFORE MII WAS CALLED.     FCCA513X    4    A
HISTORY            BEFORE FXI CALCULATED LEA.                                 FCCA513X    5    A
HISTORY            SPS    10-10-79    APPLIES TO FTN4.8,CCG.                  FCCA513X    6    A
HISTORY                                                                       FCCA513X    7    A
HISTORY   FCC2230X GPO/DIF.  DIF ASSUMED RK = 0 WAS A SUFFICIENT CONDITION FOR FCC2230X    1    A
HISTORY            RECOGNIZING UNARY TYPE I INSTRUCTION.  HOWEVER A PACK WITH FCC2230X    2    A
HISTORY            A RJ = 0 IS AN EXCEPTION.  DIF WOULD THEN PICK UP A BOS IN FCC2230X    3    A
HISTORY            FORMING AN INCREMENT FORMULA.                              FCC2230X    4    A
HISTORY            SPS    9-19-79     APPLIES TO FTN4.8,CCG.                  FCC2230X    5    A
HISTORY                                                                       FCC2230X    6    A
HISTORY   CCGA025  GPO/MIP.  AVOID MOVING MULTIPLY THAT IS NOT UNCONDITIONALLY FCC2271X    1    A
HISTORY            EXECUTED IF OPERANDS NOT SAFE.                             FCC2271X    2    A
HISTORY            MAH    12-15-79    APPLIES TO CCG.                         FCC2271X    3    A
HISTORY                                                                       FCC2271X    4    A
HISTORY   FCC2273X BDT/RIO.  BAD CODE GENERATED FOR FCC2157.  FALSE ASSUMPTION FCC2273X    1    A
HISTORY            WAS MADE BECAUSE BM FLAG TEMPORARILY SET ON BOS.  FOLLOWING FCC2273X    2    A
HISTORY            CONSEQUTIVE DEFS AT BEGINNING OF SEQUENCE - LOOP AT RIO16  FCC2273X    3    A
HISTORY            EXITED ON A BOS AND NOT A DEF AND THE RMIN VARIABLE(KEEPS  FCC2273X    4    A
HISTORY            FIRST DEF IN SEQUENCE) WAS NOT UPDATED.                    FCC2273X    5    A
HISTORY            REPLACES (PURGES) FCCA504AX.                               FCC2273X    6    A
HISTORY            SPS    12-17-79    APPLIES FTN 4.8, CCG                    FCC2273X    7    A
HISTORY                                                                       FCC2273X    8    A
HISTORY   *L528*    ************************************************
>>>>     *L528*   5                                                           *L528*      1    A
HISTORY   F1037CC  MIO.  AVOID 176 INSTRUCTIONS AT COMPILE TIME IF *MI*
>>>>     F1037CC  5                                                           F1037CC     1    A
HISTORY            COMPILER SELECTED.
>>>>     F1037CC  6                                                           F1037CC     2    A
HISTORY            MAP    12-17-79    APPLIES TO CCG .
>>>>     F1037CC  7                                                           F1037CC     3    A
HISTORY
>>>>     F1037CC  8                                                           F1037CC     4    A
HISTORY   CCGA027  CGTM. AVOID ASSORTED MODE ERRORS OR TIME LIMITS IN OPT=2.
>>>>     CCGA027  5                                                           CCGA027     1    A
HISTORY            BLOCK TABLE WAS BEING SHRUNK BY TABLE OVERFLOW PROCESSOR
>>>>     CCGA027  6                                                           CCGA027     2    A
HISTORY            ALTHOUGH TABLE MANAGER WAS INVOKED TO ALLOCATE BLOCK SPACE.
>>>>     CCGA027  7                                                           CCGA027     3    A
HISTORY            MAH    12-26-79    APPLIES TO CCG .
>>>>     CCGA027  8                                                           CCGA027     4    A
HISTORY
>>>>     CCGA027  9                                                           CCGA027     5    A
HISTORY   CCGA028  CGTM. AVOID POTENTIAL BAD CODE IN OPT=2 BECAUSE OF REGISTER
>>>>     CCGA028  5                                                           CCGA028     1    A
HISTORY            ERROR.
>>>>     CCGA028  6                                                           CCGA028     2    A
HISTORY            MAH    01-02-80    APPLIES TO CCG .
>>>>     CCGA028  7                                                           CCGA028     3    A
HISTORY

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN HISTORY

```
>>>>     CCGA028  8                                                          CCGA028        4     A
HISTORY   CCGA029  GPO/MII. AVOID MOVING INVARIANT INDEXED LOAD OUT OF
>>>>     CCGA029  5                                                          CCGA029        1     A
HISTORY            LOOP IF PROTECTED BY CONDITIONAL BRANCH, UNLESS OPT=3.
>>>>     CCGA029  6                                                          CCGA029        2     A
HISTORY            MAH    01-08-80    APPLIES TO CCG .
>>>>     CCGA029  7                                                          CCGA029        3     A
HISTORY
>>>>     CCGA029  8                                                          CCGA029        4     A
HISTORY   CCGA030  GPO/MII. AVOID MOVING INVARIANT STORE TO PARAMTER LIST
>>>>     CCGA030  5                                                          CCGA030        1     A
HISTORY            WHICH MAY BE USED BY ANOTHER ROUTINE CALL IN SAME LOOP.
>>>>     CCGA030  6                                                          CCGA030        2     A
HISTORY            MAH    01-22-80    APPLIES TO CCG .
>>>>     CCGA030  7                                                          CCGA030        3     A
HISTORY
>>>>     CCGA030  8                                                          CCGA030        4     A
HISTORY   CCGA031  MCG/UJR.  CONFLICT FOUND IN REGISTER USAGE OF X5.  AT URJ3 IF
>>>>     CCGA031  5                                                          CCGA031        1     A
HISTORY            THE TOP BIT OF X5 IS 0 THEN A SUTIABLE RESULT REGISTER HAS
>>>>     CCGA031  6                                                          CCGA031        2     A
HISTORY            BEEN FOUND.  OTHERWISE, EITHER X6 OR X7 MUST BE STORED AND UJR
>>>>     CCGA031  7                                                          CCGA031        3     A
HISTORY            TRIES TO FIND A SUTIABLE REGISTER.  THE INSTRUCTIONS REMOVED
>>>>     CCGA031  8                                                          CCGA031        4     A
HISTORY            FROM THIS CORRECTIVE CODE DESTROYED THE CONTENTS OF X5 BEFORE
>>>>     CCGA031  9                                                          CCGA031        5     A
HISTORY            THE CHECK COULD BE MADE.
>>>>     CCGA031 10                                                          CCGA031        6     A
HISTORY            SPS    80/03/12    APPLIES TO FTN 4.8, OPT.
>>>>     CCGA031 11                                                          CCGA031        7     A
HISTORY
>>>>     CCGA031 12                                                          CCGA031        8     A
HISTORY   CCGA034  MCG/SII.  REPAIR REGISTER ERROR WHICH AFFECTS PROGRAMS
>>>>     CCGA034  5                                                          CCGA034        1     A
HISTORY            WITH COMPLICATED SEQUENCES CONTAINING LEVEL 0 VARIABLES.
>>>>     CCGA034  6                                                          CCGA034        2     A
HISTORY            MAH    80/06/26    APPLIES TO CCG.
>>>>     CCGA034  7                                                          CCGA034        3     A
HISTORY
>>>>     CCGA034  8                                                          CCGA034        4     A
HISTORY   FCCA546X MCG
>>>>     FCCA546X 5                                                          FCCA546X       1     A
HISTORY            THE COMBINATION .CPU=73 AND .DAL=NZ CAUSES MCG ASSEMBLY
>>>>     FCCA546X 6                                                          FCCA546X       2     A
HISTORY            ERRORS.  THIS HAPPENS WHEN HF.LIST IN IPTEXT SELECTS L, S0,
>>>>     FCCA546X 7                                                          FCCA546X       3     A
HISTORY            AND PS.   (LCM AND A SERIAL MACHINE)
>>>>     FCCA546X 8                                                          FCCA546X       4     A
HISTORY            TMH    06/02/80    APPLIES TO FTN,OPT AND CCG
>>>>     FCCA546X 9                                                          FCCA546X       5     A
HISTORY

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN HISTORY

>>>>     FCCA546X10                                                          FCCA546X       6     A
HISTORY   FCCA550X BDT/IPL.  CLEAR EXTRANEOUS FIELDS OF LINK WORD TO AVOID
>>>>     FCCA550X 5                                                          FCCA550X       1     A
HISTORY            MODE 1 IN BDT AT OPT=1.
>>>>     FCCA550X 6                                                          FCCA550X       2     A
HISTORY            SPS    80/06/12    APPLIES TO FTN,OPT AND CCG.
>>>>     FCCA550X 7                                                          FCCA550X       3     A
HISTORY
>>>>     FCCA550X 8                                                          FCCA550X       4     A
HISTORY   *L538*    ***********************************************
>>>>     *L538*   5                                                          *L538*         1     A
HISTORY   CCG*81   CGTM.  UPDATE COPYWRITE INFORMATION FOR 1981.
>>>>     CCG*81   5                                                          CCG*81         1     A
HISTORY            MRR    80/09/26    APPLIES TO CCG.
>>>>     CCG*81   6                                                          CCG*81         2     A
HISTORY
>>>>     CCG*81   7                                                          CCG*81         3     A
HISTORY   CCGA035  GPO/RDD.  WHEN THE TRACEBACK OPTION IS SELECTED APLISTS ARE
>>>>     CCGA035  5                                                          CCGA035        1     A
HISTORY            GENERATED FOR ALL FUNCTION CALLS EVEN BASIC EXTERNAL FUNCTIONS
>>>>     CCGA035  6                                                          CCGA035        2     A
HISTORY            (BEF'S).  IF RDD ENCOUNTERS A FUNCTION CALL IN ITS SCAN OF
>>>>     CCGA035  7                                                          CCGA035        3     A
HISTORY            DEAD DEFINITIONS THE PROCESS IS USUALLY HALTED SO THAT THE
>>>>     CCGA035  8                                                          CCGA035        4     A
HISTORY            CORRESPONDING APLIST MAY BE PROPERLY GENERATED.  THE EXCEPTION
>>>>     CCGA035  9                                                          CCGA035        5     A
HISTORY            IS FOR BEF FUNCTION CALLS BECAUSE IT IS ASSUMED THAT ALL
>>>>     CCGA035 10                                                          CCGA035        6     A
HISTORY            PARAMETERS ARE PASSED USING REGISTERS.  THIS IS NOT THE CASE
>>>>     CCGA035 11                                                          CCGA035        7     A
HISTORY            WHEN THE TRACEBACK OPTION IS SELECTED.  CODE HAS BEEN ADDED TO
>>>>     CCGA035 12                                                          CCGA035        8     A
HISTORY            CHECK IF THE TRACEBACK OPTION WAS SELECTED AND IF SO TO HALT
>>>>     CCGA035 13                                                          CCGA035        9     A
HISTORY            THE PROCESS OF KILLING DEAD DEFINITIONS.
>>>>     CCGA035 14                                                          CCGA035       10     A
HISTORY            REQUIRES FC5A275.
>>>>     CCGA035 15                                                          CCGA035       11     A
HISTORY            SPS/MRR 80/08/28   APPLIES TO CCG.
>>>>     CCGA035 16                                                          CCGA035       12     A
HISTORY
>>>>     CCGA035 17                                                          CCGA035       13     A
HISTORY   CCGA036  GRA/MFA.  FCC2135 WAS INTRODUCED TO CORRECT A PROBLEM WITH
>>>>     CCGA036  5                                                          CCGA036        1     A
HISTORY            LOOP TEST MECHANISMS.  THE OPTIMIZER IN SEEKING TO REDUCE THE
>>>>     CCGA036  6                                                          CCGA036        2     A
HISTORY            NUMBER OF REGISTERS IT ASSIGNS ACROSS A LOOP SEEKS TO REPLACE
>>>>     CCGA036  7                                                          CCGA036        3     A
HISTORY            THE LOOP VARIABLE WITH A VARIANT ASSIGNED TO A B REGISTER
>>>>     CCGA036  8                                                          CCGA036        4     A
HISTORY            WHICH INDEXES AN ARRAY IN THE LOOP TEST MECHANISM.  HOWEVER

                 0         1         2         3         4         5         6         7         8
                 12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN HISTORY

```
>>>>     CCGA036  9                                                          CCGA036        5     A
HISTORY            TROUBLE WITH B REGISTER OVERFLOW OCCURRED WHEN THE VARIABLE
>>>>     CCGA036 10                                                          CCGA036        6     A
HISTORY            INDEXED LCM, A FORMAL PARAMETER, OR BLANK COMMON AND THE
>>>>     CCGA036 11                                                          CCGA036        7     A
HISTORY            INCREMENT VALUE WAS A SUFFICIENTLY LARGE CONSTANT.  FCC2135
>>>>     CCGA036 12                                                          CCGA036        8     A
HISTORY            INTRODUCED CODE TO CHOOSE AN ALTERNATIVE MECHANISM IF ANY OF
>>>>     CCGA036 13                                                          CCGA036        9     A
HISTORY            THE ABOVE CONDITIONS WERE MET.  HOWEVER PROBLEMS ALSO OCCURRED
>>>>     CCGA036 14                                                          CCGA036       10     A
HISTORY            WHEN THE OPTION LCM=I IS SELECTED.  IN THIS CASE THE RLIST
>>>>     CCGA036 15                                                          CCGA036       11     A
HISTORY            CODE GENERATED IS SUCH THAT THERE IS NO EASY WAY OF DETERMININ
>>>>     CCGA036 16                                                          CCGA036       12     A
HISTORY            WHAT ARRAYS THE VARIANT IS REALLY INDEXING.  CODE HAS BEEN
>>>>     CCGA036 17                                                          CCGA036       13     A
HISTORY            INTRODUCED SO THAT THE VARIANT I WILL NOT BE USED IN LOOP TEST
>>>>     CCGA036 18                                                          CCGA036       14     A
HISTORY            REPLACEMENT IF THERE IS A RCT ENTRY SUCH THAT RF=I AND IH=0.
>>>>     CCGA036 19                                                          CCGA036       15     A
HISTORY            SPS    80/08/28    APPLIES TO CCG.
>>>>     CCGA036 20                                                          CCGA036       16     A
HISTORY
>>>>     CCGA036 21                                                          CCGA036       17     A
HISTORY   CCGA037  CGTM/WII.  SIGN EXTENSION CAUSED 18 BIT ARITHMETIC
>>>>     CCGA037  5                                                          CCGA037        1     A
HISTORY            ERROR.  THIS PREVENTED THE MAT BIT IN THE SYMBOL
>>>>     CCGA037  6                                                          CCGA037        2     A
HISTORY            TABLE FROM BEING SET.  THIS CAUSED A REFERENCED
>>>>     CCGA037  7                                                          CCGA037        3     A
HISTORY            VARIABLE TO GO UNALLOCATED.
>>>>     CCGA037  8                                                          CCGA037        4     A
HISTORY            SPS    80/09/05    APPLIES TO CCG.
>>>>     CCGA037  9                                                          CCGA037        5     A
HISTORY
>>>>     CCGA037 10                                                          CCGA037        6     A
HISTORY   CCGA039  SQZ/SBB, GRA/IRA,CUC.  CCG170 FAILS TO RECOGNIZE LEVEL 0
>>>>     CCGA039  5                                                          CCGA039        1     A
HISTORY            FORMAL PARAMETERS AT SEVERAL PLACES AND THUS GENERATES
>>>>     CCGA039  6                                                          CCGA039        2     A
HISTORY            CODE FOR LEVEL 2 FORMAL PARAMETERS.
>>>>     CCGA039  7                                                          CCGA039        3     A
HISTORY            SPS    81/01/22    APPLIES TO CCG.
>>>>     CCGA039  8                                                          CCGA039        4     A
HISTORY   *L552*    ************************************************       *L552*         1     A
HISTORY   CCGA040  PROSEQ/ESR.  FAILURE TO RECOGNIZE LEVEL 0 REFERENCES ON  CCGA040        1     A
HISTORY            STT INSTRUCTIONS RESULTED IN BAD CODE WHEN LCM=D WAS CHOSEN.  CCGA040        2     A
HISTORY            SPS 81/03/23       APPLIES TO CCG.                       CCGA040        3     A
HISTORY                                                                     CCGA040        4     A
HISTORY   CCGA041  CGTM/WII.  FIX LINE/SEQUENCE MAGNITUDE TEST FOR TRACE WORD  CCGA041        1     A
HISTORY            ON RJ6 INSTRUCTIONS.  THIS REQUIRES CODE FC50264 FOR CORRECT  CCGA041        2     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN HISTORY

```
HISTORY            COMPILE TIME CODE PRODUCTION AND FL50429 FOR RUN TIME SUPPORT.        CCGA041    3    A
HISTORY            MRR    81/04/04    APPLIES TO CCG.                                    CCGA041    4    A
HISTORY                                                                                   CCGA041    5    A
HISTORY   CCGA042  GPO/SLC.  ATTEMPTS TO REPLACE A LOAD OF A LEVEL 2                      CCGA042    1    A
HISTORY            FORMAL PARAMETER WHICH IS ALSO AN IMMEDIATE PREDECESSOR OF A           CCGA042    2    A
HISTORY            STORE WITH A LOAD OF A LOCAL COPY CAUSES MAYHEM IN MCG.                CCGA042    3    A
HISTORY            SPS    81/04/03    APPLIES TO CCG.                                     CCGA042    4    A
HISTORY                                                                                   CCGA042    5    A
HISTORY   CCGA043  GPO/MPR.  CORRECTLY ACCESS THE FP.CA FIELD.  SUPPORT CODE              CCGA043    1    A
HISTORY            FOR FC50252, WHICH ALLOWS CORRECT CODE GENERATION OF LOCAL             CCGA043    2    A
HISTORY            COPIES OF FORMAL PARAMETERS.                                           CCGA043    3    A
HISTORY            MRR    81/04/14    APPLIES TO CCG.                                     CCGA043    4    A
HISTORY                                                                                   CCGA043    5    A
HISTORY   CCGA044  PROSEQ/ESR.  WHEN EXPANDING A LCM REFERENCE WITH THE LCM=G             CCGA044    1    A
HISTORY            OPTION SELECTED, ESR ATTEMPTED TO ESTABLISH THE OFFSET VIA A           CCGA044    2    A
HISTORY            18 BIT SET INSTRUCTION.  A NEGATIVE OFFSET RESULTS WHEN THE            CCGA044    3    A
HISTORY            OFFSET EXCEEDS THE 17 BIT CAPACITY.  CODE WAS ADDED TO RECOG-          CCGA044    4    A
HISTORY            NIZE SUCH LARGE OFFSETS AND MATERIALIZE THEM VIA A FULL WORD           CCGA044    5    A
HISTORY            LOAD FROM THE CONSTANT TABLE.                                          CCGA044    6    A
HISTORY            SPS    81/05/05    APPLIES TO CCG.                                     CCGA044    7    A
HISTORY                                                                                   CCGA044    8    A
HISTORY   CCGA045  SQZ/MSL.  WHEN PROCESSING CONDITIONAL JUMPS OF THE TYPE PL/MI,         CCGA045    1    A
HISTORY            SQZ FIRST EXAMINES WHETHER THE RI OF THE JUMP IS THE RESULT OF         CCGA045    2    A
HISTORY            A XMTC.  IF SO THEN THE JUMP INSTRUCTION IS TOGGLED AND THE RJ         CCGA045    3    A
HISTORY            OF THE XMTC BECOMES THE NEW RI OF THE TOGGLED INSTRUCTION.  IF         CCGA045    4    A
HISTORY            THE RI OF THE JUMP INSTRUCTION IS A CONSTANT, THEN EITHER AN           CCGA045    5    A
HISTORY            UNCONDITIONAL JUMP REPLACES THE INSTRUCTION OR THE INSTRUCTION         CCGA045    6    A
HISTORY            IS ELIMINATED ALTOGETHER DEPENDING ON THE VALUE OF RI.  A              CCGA045    7    A
HISTORY            PROBLEM OCCURRED WHEN THE INSTRUCTION WAS TOGGLED AS DESCRIBED         CCGA045    8    A
HISTORY            ABOVE.  THE OLD RI VALUE WAS EXAMINED INSTEAD OF THE NEW RI OF         CCGA045    9    A
HISTORY            THE TOGGLED INSTRUCTION.  AS A RESULT, THE GENERATED CODE WAS          CCGA045   10    A
HISTORY            JUST THE REVERSE AS TO WHAT WAS DESIRED.                               CCGA045   11    A
HISTORY            SPS    81/05/07    APPLIES TO CCG.                                     CCGA045   12    A
HISTORY                                                                                   CCGA045   13    A
HISTORY   CCGA046  GPO/MIP.  A PROBLEM OCCURS WHEN A XMTC IS INCLUDED IN INTEGER          CCGA046    1    A
HISTORY            POLYNOMIALS (I.P.'S) THAT ARE INCLUDED IN STRENGTH REDUCTION.          CCGA046    2    A
HISTORY            IN A PARTICULAR CIRCUMSTANCE, THE PROPER EXECUTION OF CODE             CCGA046    3    A
HISTORY            INSIDE A LOOP DEPENDS UPON THE VALUE OF THE I.P. BEING  (-0)           CCGA046    4    A
HISTORY            DURING A CERTAIN ITERATION OF THE LOOP.  HOWEVER, WHEN THE             CCGA046    5    A
HISTORY            XMTC IS MOVED OUTSIDE OF THE LOOP AND REPLACED WITH STRENGTH           CCGA046    6    A
HISTORY            REDUCTION CODE, THE VALUE OF THE I.P. IS (+0) ON THE CRITICAL          CCGA046    7    A
HISTORY            ITERATION.  IT WAS DECIDED NOT TO INCLUDE XMTC IN I.P.'S.              CCGA046    8    A
HISTORY            SPS    81/05/13    APPLIES TO FTN4,CCG.                                CCGA046    9    A
HISTORY            NOTE: THIS FIX HAS NOT BEEN PROPAGATED TO FTN4.                        CCGA046   10    A
HISTORY                                                                                   CCGA046   11    A
HISTORY   CCGA047  SQZ/MSL.  A MISALIGNED RN ENTRY CAUSED THE CONSTANT BIT                CCGA047    1    A
HISTORY            TO BE IMPROPERLY SET.  THIS IN TURN RESULTED IN ONE OF JASIK'S         CCGA047    2    A
HISTORY            UNFORSEEN SITUATIONS.                                                  CCGA047    3    A
HISTORY            SPS    81/05/21    APPLIES TO CCG.                                     CCGA047    4    A
HISTORY                                                                                   CCGA047    5    A
HISTORY   CCG0001  CGIA.  CORRECTLY FORM THE TARGET MACHINE MICRO.  PROVIDES              CCG0001    1    A
HISTORY            CORRECT TARGET MACHINE SETTING IN 7700 LOADER TABLE.                   CCG0001    2    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN HISTORY

```
HISTORY            MRR    81/01/16    APPLIES TO CCG.                                  CCG0001        3     A
HISTORY                                                                                CCG0001        4     A
HISTORY   CCG0004  PROSEQ/ESR.  WHEN LCM=G IS CHOSEN, A BUG IN ESR BYPASSES CODE        CCG0004        1     A
HISTORY            TO CHECK WHETHER THE SPECIAL REFERENCE BEING EXPANDED PRECEDES       CCG0004        2     A
HISTORY            A REGISTER SPECIFY TO A NON-X REGISTER.  AS A RESULT ESR EMITS       CCG0004        3     A
HISTORY            A XMT INSTEAD OF THE NECESSARY SA.                                   CCG0004        4     A
HISTORY            SPS    81/05/14    APPLIES TO CCG.                                   CCG0004        5     A
HISTORY                                                                                CCG0004        6     A
HISTORY   *L564*    ************************************************                   *L564*        1     A
HISTORY   CCG*82   CGTM.  UPDATE COPYWRITE INFORMATION FOR 1982.                        CCG*82         1     A
HISTORY            MRR    81/09/10    APPLIES TO CCG.                                   CCG*82         2     A
HISTORY                                                                                CCG*82         3     A
HISTORY   CCGA049  CCGTEXT,SQZ/STS,GPO                                                  CCGA049        1     A
HISTORY            THE CA FIELD IN THE TEMPORARY EQUIVALENCE TABLE (TET)                CCGA049        2     A
HISTORY            IS USED FOR TWO PURPOSES. THE OPT=2 COMPILER GENERATES               CCGA049        3     A
HISTORY            LOADS OF THE WRONG TEMP BECAUSE SQZ DESTROYS THE TET INDEX           CCGA049        4     A
HISTORY            IN THE CA FIELD WHEN IT ATTEMPTS TO EQUIVALENCE TWO ASSIGNED         CCGA049        5     A
HISTORY            TEMPS. THE FIX IS TO PREVENT THE EQUIVALENCE UNTIL AFTER             CCGA049        6     A
HISTORY            THE TEMP LOAD HAS BEEN GENERATED AND THE CA FIELD IS FREE            CCGA049        7     A
HISTORY            TO CONTAIN A TEMP NUMBER.                                            CCGA049        8     A
HISTORY            SEE FCC2392 IN FORTRAN 4.                                            CCGA049        9     A
HISTORY            TMH    81/07/14    APPLIES TO CCG,OPT=2.                             CCGA049       10     A
HISTORY                                                                                CCGA049       11     A
HISTORY   CCGA052  PROSEQ/ISC.  WHEN EXPANDING A LCM REFERENCE WITH THE LCM=G           CCGA052        1     A
HISTORY            OPTION SELECTED, ESR ATTEMPTED TO ESTABLISH THE OFFSET VIA A         CCGA052        2     A
HISTORY            18 BIT SET INSTRUCTION.  A NEGATIVE OFFSET RESULTS WHEN THE          CCGA052        3     A
HISTORY            OFFSET EXCEEDS THE 17 BIT CAPACITY.  CODE WAS ADDED TO RECOG-        CCGA052        4     A
HISTORY            NIZE SUCH LARGE OFFSETS AND MATERIALIZE THEM VIA A FULL WORD         CCGA052        5     A
HISTORY            LOAD FROM THE CONSTANT TABLE.  REPAIRS CCGA044.                      CCGA052        6     A
HISTORY            SIJ/MRR 81/09/11   APPLIES TO CCG.                                   CCGA052        7     A
HISTORY                                                                                CCGA052        8     A
HISTORY   CCGA054  MOVE WEORS IN CCG TO THEIR OWN DECKS AS PER INTEGRATION              CCGA054        1     A
HISTORY            REQUIREMENT. ALSO CONVERTS TESTMODE DEF TO *TEST*.                   CCGA054        2     A
HISTORY            SCC    10/19/81    CMPLTXT, WEOR1*, CCGTEXT, WEOR2*                  CCGA054        3     A
HISTORY                                                                                CCGA054        4     A
HISTORY   CCG0006  SQZ/STT.  DONT SET RF FIT FOR STT INTRUCTIONS WITH LEVEL 2 AND       CCG0006        1     A
HISTORY            LCM=G.                                                               CCG0006        2     A
HISTORY            SIJ/MRR 81/08/07   APPLIES TO CCG.                                   CCG0006        3     A
HISTORY                                                                                CCG0006        4     A
HISTORY   CCG0013  GPO/GPO.  CORRECT TABLE POINTER ACCESS IN GPO. MAKE                  CCG0013        1     A
HISTORY            ACCESS INDIRECT.  PREVENTS INADVERTENT FREE SPACE ALLOCATION         CCG0013        2     A
HISTORY            IN BLK BY ALLOC.                                                     CCG0013        3     A
HISTORY            SIJ    81/11/17    APPLIES TO CCG, OPT=2.                            CCG0013        4     A
HISTORY                                                                                CCG0013        5     A
HISTORY   *L577*    ************************************************                   *L577*        1     A
HISTORY   CCGA062  GRA/IRA/PVC.  WHEN OPERATION IS [S RI,-CON] AND [CON] IS IN A         CCGA062        1     A
HISTORY            B-REG, DO NOT SET B-REG FIELD IN THE LINK WORD.                      CCGA062        2     A
HISTORY            SIJ    81/12/07    APPLIES TO CCG.                                   CCGA062        3     A
HISTORY                                                                                CCGA062        4     A
HISTORY   CCGA063  GRA/MFA.  TEST INV BIT IN TET WORD BEFORE TESTING ITI FIELD           CCGA063        1     A
HISTORY            SINCE THE LATTER MAY BE NON-ZERO IN AN OUTER LOOP, AS ITI AND        CCGA063        2     A
HISTORY            HBN OCCUPY THE SAME BITS IN THE WORD.                                CCGA063        3     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN HISTORY

```
HISTORY            SIJ    81/12/08    APPLIES TO CCG.                                  CCGA063        4     A
HISTORY                                                                                CCGA063        5     A
HISTORY   CCGA066  MCG/ASC.  CORRECT SHIFT INSTRUCTION IN CODE FOR 175 MACHINE.         CCGA066        1     A
HISTORY            PREVENTS PLI FAILURE.                                               CCGA066        2     A
HISTORY            SIJ    81/12/23    APPLIES TO CCG, 175.                             CCGA066        3     A
HISTORY                                                                                CCGA066        4     A
HISTORY   CCGA067  CGIA/VFDP.  REVISE VFDP PROCESSING TO PUT OUT A N WORD VFDP          CCGA067        1     A
HISTORY            TABLE. THIS VERSION ACCEPTS ARBITRARY LONG BIT STRINGS.             CCGA067        2     A
HISTORY            SIJ    81/12/28    APPLIES TO CCG, PLI.                             CCGA067        3     A
HISTORY                                                                                CCGA067        4     A
HISTORY   CCGA068  CGIA.  FIX THE FOLLOWING CCG INTERNAL ASSEMBLER PROBLEMS:            CCGA068        1     A
HISTORY            1.  FIX BUG IN USE BLOCK SWITCHING  WHERE FIRST INST IN A TEXT      CCGA068        2     A
HISTORY                TABLE WAS LOST WITH REPEATED BLOCK SWITCHS.                     CCGA068        3     A
HISTORY            2.  FIX BUG IN ORG PROCESSING, WRONG SHIFT COUNT TO DETECT          CCGA068        4     A
HISTORY                RL=1.                                                           CCGA068        5     A
HISTORY            3.  FIX BUG IN DTT AND APT WHERE TEXT TABLE LENGTH IS NOT SET       CCGA068        6     A
HISTORY                PROPERLY.                                                       CCGA068        7     A
HISTORY            4.  SUBSTITUTE PROPER CA IN STT INSTRUCTIONS THAT REF CON.          CCGA068        8     A
HISTORY            SIJ    81/12/28    APPLIES TO CCG, PLI.                             CCGA068        9     A
HISTORY                                                                                CCGA068       10     A
HISTORY   CCGA072  CCGTEXT,CGIA/UJP/USEBLK/USE, CGTM/CONTV/CG$RBT/CG$EP.               CCGA072        1     A
HISTORY            ALLOW LARGER NUMBER OF LOCAL BLOCKS IN OBJECT PROGRAM.              CCGA072        2     A
HISTORY            THIS IDENT REQUIRES PLIA463 AND FC5A399 FOR ENTRY POINTS.           CCGA072        3     A
HISTORY            SIJ    82/04/21    APPLIES TO CCG, PL/I.                            CCGA072        4     A
HISTORY                                                                                CCGA072        5     A
HISTORY   CCGA073  MCG/SII/LSS/PLD.  ALLOW R-NUMBER 1 TO DEFINE A0 OR B7 AS SET        CCGA073        1     A
HISTORY            SET BY CALLER USED BY FPLI.  THIS MOD IS NOT LINKED INTO THE        CCGA073        2     A
HISTORY            LOOP OPTIMIZER OR AIS AND INCORRECT RESULTS MAY RESULT IF A         CCGA073        3     A
HISTORY            CODE SEQUENCE GOES THROUGH EITHER OF THEM WHEN REG=A0 IS SET        CCGA073        4     A
HISTORY            TO B7 .                                                             CCGA073        5     A
HISTORY            SIJ    82/04/21    APPLIES TO CCG.                                  CCGA073        6     A
HISTORY                                                                                CCGA073        7     A
HISTORY   CCG0002  SQZ/ECO/RJXJ.  PROVIDE CONSTANT REDUCTION FOR SS INSTRUCTION.       CCG0002        1     A
HISTORY            WHEN PROCESSING PLI RJ, GENERATE RJ3 [ NOT RJ6 ].                   CCG0002        2     A
HISTORY            SIJ    81/12/16    APPLIES TO CCG.                                  CCG0002        3     A
HISTORY                                                                                CCG0002        4     A
HISTORY   CCG0008  MCG/UJR.  FIX CASE WHERE NO REGISTERS AVAILABLE, OPERANDS ARE        CCG0008        1     A
HISTORY            IN X6 AND X7 AND RESULT PRECEDES A REGISTER STORE BY SAVING         CCG0008        2     A
HISTORY            X6/X7, FREEING THE RESULT REGISTER AND RELOADING THE OPERAND.       CCG0008        3     A
HISTORY            SIJ    82/02/04    APPLIES TO CCG.                                  CCG0008        4     A
HISTORY                                                                                CCG0008        5     A
HISTORY   CCG0018  GRA/AAV.  REDUCE MAA BY PROPER AMOUNT SO DAA DOESNT ATTEMPT TO       CCG0018        1     A
HISTORY            ASSIGN MORE CANDIDATES THAN REMAINING LOAD REGISTERS.               CCG0018        2     A
HISTORY            SIJ    81/12/28    APPLIES TO CCG.                                  CCG0018        3     A
HISTORY                                                                                CCG0018        4     A
HISTORY   CCG0022  GRA/GRA.  CLEAR NBIP# SO BDT DOESN'T CHECK IT.  PREVENTS             CCG0022        1     A
HISTORY            INFINITE LOOP AT OPT=1.                                             CCG0022        2     A
HISTORY            SIJ    81/12/22    APPLIES TO CCG, OPT=1.                           CCG0022        3     A
HISTORY                                                                                CCG0022        4     A
HISTORY   *L587*    ************************************************                   *L587*        1     A
HISTORY   CCGA076  FA=WTH.  ADD COMDECK FA=WTH.  PROVIDES SCOPE 2 OUTPUT WITH          CCGA076        1     A
HISTORY            TRAILING BLANK SUPRESSION.  THIS IDENT REQUIRES CONCURRENT          CCGA076        2     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN HISTORY

```
HISTORY            INSTALLATION OF FC50467.                                  CCGA076    3    A
HISTORY            PURGES=FA=WTC                                             CCGA076    4    A
HISTORY            MRR    82/07/21    FA=WTC, FA=WTH, FA=WTW.                CCGA076    5    A
HISTORY                                                                      CCGA076    6    A
HISTORY   CCGA079  FA=OPE.  ALLOW REW=B TO WORK ON SCOPE 2.                  CCGA079    1    A
HISTORY            REQUIRES CONCURRENT INSTALLATION OF FC50681.              CCGA079    2    A
HISTORY            DEPENDENCY=CCGA018                                        CCGA079    3    A
HISTORY            MRR    82/11/19    FA=OPE                                 CCGA079    4    A
HISTORY                                                                      CCGA079    5    A
HISTORY   CCGA082  GRA/DUM.  CLEAR INC BIT IN DESCR OF PRED OF PST SO THAT ANY CCGA082    1    A
HISTORY            PRED OF AN INTEGER ADD OR SUBTRACT IS MATERIALIZED IN AN  CCGA082    2    A
HISTORY            X-REGISTER.                                               CCGA082    3    A
HISTORY            SIJ    82/12/17    GRA                                    CCGA082    4    A
HISTORY                                                                      CCGA082    5    A
HISTORY   CCG0037  GRA/DUM.  SET EU BIT FOR XMTC TO AVOID PROBLEM WITH SUBSCRIPT CCG0037    1    A
HISTORY            OF FORM [ARRAY(CON-I)] IN LOOP.                           CCG0037    2    A
HISTORY            SIJ    82/04/28    GRA                                    CCG0037    3    A
HISTORY                                                                      CCG0037    4    A
HISTORY   CCG0041  CCGILFD,SQZ/CTI - REFERENCES TO LEVEL 0 DUMMY ARGUMENTS   CCG0041    1    A
HISTORY            ARE INCORRECTLY GENERATED IF THE TARGET MACHINE HAS       CCG0041    2    A
HISTORY            LCM.  CODE FOR SCM IS GENERATED USING A STORE REGISTER    CCG0041    3    A
HISTORY            FOR A FETCH.  IF THE *XMT* BIT IS SET IN THE DESCRIPTOR   CCG0041    4    A
HISTORY            WORD OF A *LD*, THAT INSTRUCTION WILL NO LONGER BE        CCG0041    5    A
HISTORY            ELIMINATED.                                               CCG0041    6    A
HISTORY            AXM    82/04/26    CCGILFD, SQZ                           CCG0041    7    A
HISTORY                                                                      CCG0041    8    A
HISTORY   CCG0045  GRA/IRA - FIX R-NUMBER ASSIGNMENTS FOR LEVEL 0            CCG0045    1    A
HISTORY            *DRL* INSTRUCTIONS.  DO NOT SET R-NUMBER IN ST-REG        CCG0045    2    A
HISTORY            FIELD IF LEVEL 0 BIT SET IN DESCRIPTOR.                   CCG0045    3    A
HISTORY            AXM    82/06/02    GRA                                    CCG0045    4    A
HISTORY                                                                      CCG0045    5    A
HISTORY   CCG0071  GRA/SIV.  IF FTN5 HAS ARG=COMMON SET, REMOVE UPPER BITS   CCG0071    1    A
HISTORY            FROM AP-LISTS.                                            CCG0071    2    A
HISTORY            MOD SET *FC50818* IN FTN5 MUST BE PRESENT.                CCG0071    3    A
HISTORY            DEPENDENCY=FCC0912                                        CCG0071    4    A
HISTORY            SIJ    83/01/14    GRA                                    CCG0071    5    A
HISTORY                                                                      CCG0071    6    A
HISTORY   CCG0085  PROSEQ/ISC.  PROVIDE MINUS ZERO PROTECT FOR RF+CA .NE. 0 TEST. CCG0085    1    A
HISTORY            PREVENTS BAD CODE FOR LCM=I/G ADDRESS CALCULATIONS.       CCG0085    2    A
HISTORY            MRR/SIJ 82/12/14   PROSEQ                                 CCG0085    3    A
HISTORY                                                                      CCG0085    4    A
HISTORY   *L601*    ************************************************         *L601*     1    A
HISTORY   CCGA083  SQZ/RJ3/GTL/GXI/MMR/IXI.  SQUEEZE BASIC EXTERNAL FUNCTION CCGA083    1    A
HISTORY            CALLS.                                                    CCGA083    2    A
HISTORY            DEPENDENCY=CCG17,CCG7                                     CCGA083    3    A
HISTORY            SIJ    83/02/17    SQZ                                    CCGA083    4    A
HISTORY                                                                      CCGA083    5    A
HISTORY   CCG0012  GRA/IRA/XMT, SQZ/SMR.  FIX *GRA* TO NOT SET ST VALUE IN LINK CCG0012    1    A
HISTORY            WORD IF PRED OF XMT PRECEEDS A REGISTER STORE (APPLIES TO CCG0012    2    A
HISTORY            OPT=2).  FIX REGISTER ERROR IN *SQZ* WHICH CAUSED DIFFERENT CCG0012    3    A
HISTORY            CODE IN MI AND NON-MI COMPILERS.                          CCG0012    4    A
HISTORY            DEPENDENCY=FCC1740X,CCG21                                 CCG0012    5    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN HISTORY

```
HISTORY            SIJ    83/05/06    GRA, SQZ                                    CCG0012        6     A
HISTORY                                                                           CCG0012        7     A
HISTORY   CCG0034  GRA/IRA.  INCREMENT OF NEGATIVE CON IN B-REG IS                CCG0034        1     A
HISTORY            NOT HANDLED CORRECTLY.                                         CCG0034        2     A
HISTORY            DEPENDENCY=CCGA062                                             CCG0034        3     A
HISTORY            SIJ    83/04/01    APPLIES TO OPT=1,2                          CCG0034        4     A
HISTORY                                                                           CCG0034        5     A
HISTORY   CCG0057  GRA/AAV/IRP.  REDUCE MAA WHEN ASSIGNING A/X REGS AND           CCG0057        1     A
HISTORY            N.LOADS GT 5.  PREVENTS INFINITE LOOP IN MCG AT OPT=2.         CCG0057        2     A
HISTORY            DEPENDENCY=CCG25,CCG34                                         CCG0057        3     A
HISTORY            SIJ    83/04/01    APPLIES TO OPT=2                            CCG0057        4     A
HISTORY                                                                           CCG0057        5     A
HISTORY   CCG0086  MCG/PRS. SUPPRESS GENERATION OF A 15 BIT SB0  B2. THE          CCG0086        1     A
HISTORY            INSTRUCTION IS ISSUED FOR TRACEBACK INFORMATION AND WITH       CCG0086        2     A
HISTORY            ONE EXCEPTION IS A 30 BIT INSTRUCTION OF THIS FORM(SB0 B2+     CCG0086        3     A
HISTORY            A NON ZERO LINE NUMBER) AND POSES NO PROBLEM. THE EXCEPTION    CCG0086        4     A
HISTORY            IS WHEN NO LINE NUMBER IS ASSOCIATED WITH A SECTION OF CODE    CCG0086        5     A
HISTORY            WHICH RESULTS IN LINE ZERO AND A 15 BIT INSTRUCTION.           CCG0086        6     A
HISTORY            THIS 15 BIT INSTRUCTION IS A READ CM ON 170/8XX MACHINES.      CCG0086        7     A
HISTORY            DEPENDENCY=FEAT155A                                            CCG0086        8     A
HISTORY            SSK    83/01/11    MCG                                         CCG0086        9     A
HISTORY                                                                           CCG0086       10     A
HISTORY   CCG0088  PROSEQ/ESR - SAVE O.MOD PROPERLY BEFORE CALL TO CON SCANNER    CCG0088        1     A
HISTORY            SO IF THE TABLE MANAGER MOVES THE TABLES, WE CAN RESTORE       CCG0088        2     A
HISTORY            A7 AND NOT BLOW UP.                                            CCG0088        3     A
HISTORY            **** NOTE - THE FOLLOWING IDENTS (ALL AVAILABLE VIA SOLVER)    CCG0088        4     A
HISTORY            **** SHOULD BE OBTAINED IN CONJUNCTION WITH THIS ONE --        CCG0088        5     A
HISTORY              FC50686 (LEVEL 577) - FORMAL PARAM CELL REINITIALIZATION.    CCG0088        6     A
HISTORY              FC50757 (LEVEL 588) - OVERALL EQUIVALENCE RE-WORK.           CCG0088        7     A
HISTORY              FC5A319 (LEVEL 588) - ADDITIONAL EQUIVALENCE CODE.           CCG0088        8     A
HISTORY              FC5A416 (LEVEL 588) - HANDLE 8-DIGIT LCM ADDRS IN MAP.       CCG0088        9     A
HISTORY              FC50825 (CURRENT LEVEL) - OPT=0 CODE.                        CCG0088       10     A
HISTORY            DEPENDENCY=CCGA044                                             CCG0088       11     A
HISTORY            SIJ    83/01/25    APPLIES TO CCG, OPT=2.                      CCG0088       12     A
HISTORY                                                                           CCG0088       13     A
HISTORY   CCG0089  GRA/IRA/JPX - ENSURE THAT B-REG VALUE IS NOT REDEFINED         CCG0089        1     A
HISTORY            BEFORE TRYING TO USE IT.                                       CCG0089        2     A
HISTORY            DEPENDENCY=CCG24                                               CCG0089        3     A
HISTORY            SIJ    83/03/11    APPLIES TO OPT=2                            CCG0089        4     A
HISTORY                                                                           CCG0089        5     A
HISTORY   *L617*    ************************************************             *L617*         1     A
HISTORY   CCG0093  GRA/AAV.  DO NOT ASSIGN A/X VARS IN OPT=1.                     CCG0093        1     A
HISTORY            DEPENDENCY=CCG25                                               CCG0093        2     A
HISTORY            SIJ    83/09/19    APPLIES TO FTN5  OPT=1                      CCG0093        3     A
HISTORY                                                                           CCG0093        4     A
HISTORY   CCG0110  MCG/UJR.  SPECIAL CASE BINARY OP THAT PRS AND                  CCG0110        1     A
HISTORY            BOTH OPERANDS HAVE ONE USE AND REGS CLOGGED TO ISSUE           CCG0110        2     A
HISTORY            INSTRUCTION FOLLOWED BY TWO XMTS TO FIX REG ASSIGNMENTS.       CCG0110        3     A
HISTORY            DEPENDENCY=CCG19,FCC0641                                       CCG0110        4     A
HISTORY            SIJ    83/09/30    APPLIES TO FTN 5.1                          CCG0110        5     A
HISTORY                                                                           CCG0110        6     A
HISTORY   CCG0117  GRA/AAV.  DONT INCREMENT NAB WHEN INDEX IS NOT ASSIGNED TO     CCG0117        1     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN HISTORY

```
HISTORY            A B-REGISTER.                                                CCG0117    2    A
HISTORY            DEPENDENCY=CCG25                                             CCG0117    3    A
HISTORY            SIJ    83/09/19    APPLIES TO FTN5, OPT=2                    CCG0117    4    A
HISTORY                                                                         CCG0117    5    A
HISTORY   CCG0119  GRA/AAV.  NO A-REGISTER PREFETCH IF ADDRESS REF NOT ON       CCG0119    1    A
HISTORY            ARTICULATION OR LATCHING NODE IN OPT=2.                      CCG0119    2    A
HISTORY            DEPENDENCY=CCG25                                             CCG0119    3    A
HISTORY            SIJ    83/09/19    APPLIES TO FTN5, OPT=2                    CCG0119    4    A
HISTORY                                                                         CCG0119    5    A
HISTORY   CCG0124  SQZ/CCV.  SET UDT INDEX OF *LDC* PROPERLY TO AVOID RANDOM BIT CCG0124    1    A
HISTORY            SETTING IN GRA WHICH CAUSED AN INDEX REG PRELOAD TO BE LOST. CCG0124    2    A
HISTORY            DEPENDENCY=CCG17                                             CCG0124    3    A
HISTORY            SIJ    83/09/21    APPLIES TO FTN5, OPT=1,2                  CCG0124    4    A
HISTORY                                                                         CCG0124    5    A
HISTORY   *L628*    ************************************************            *L628*     1    A
HISTORY   CCGA085  SQZ/LDC.  DO NOT SQUEEZE LDC IF BOTH PS AND PRS SET.         CCGA085    1    A
HISTORY            PREVENTS INCORRECT ASSIGNMENT OF A0 AS A STORE REGISTER.     CCGA085    2    A
HISTORY            DEPENDENCY=CCG17                                             CCGA085    3    A
HISTORY            SIJ    84/02/24    SQZ                                       CCGA085    4    A
HISTORY                                                                         CCGA085    5    A
HISTORY   CCGA088  GPO/MPR.  TURN OFF XMT BIT IN LEVEL 0 LOAD WHEN CHANGING IT  CCGA088    1    A
HISTORY            TO AN LC.                                                    CCGA088    2    A
HISTORY            DEPENDENCY=CCG23                                             CCGA088    3    A
HISTORY            SIJ    84/06/18    APPLIES TO FTN5, OPT=2,3                  CCGA088    4    A
HISTORY                                                                         CCGA088    5    A
HISTORY   CCG0127  GPOCOM, GPO/ATT/EIE.  DO NOT CHANGE CA FIELD OF TET ENTRY    CCG0127    1    A
HISTORY            THAT IS EQV.  ALSO, IMPROVE PROCESSING OF COMPLICATED        CCG0127    2    A
HISTORY            SUBSCRIPTS SO THAT REDUNDANT GLOBAL TEMPS ARE NOT CREATED.   CCG0127    3    A
HISTORY            APPLIES TO FTN5, OPT=2.                                      CCG0127    4    A
HISTORY            DEPENDENCY=FCC0514,FCC0717                                   CCG0127    5    A
HISTORY            SIJ    84/01/17    GPOCOM, GPO                               CCG0127    6    A
HISTORY                                                                         CCG0127    7    A
HISTORY   CCG0128  GRA/IRA/ST.  FIX PROBLEM WITH SEQUENCE OF THE FORM           CCG0128    1    A
HISTORY            OLD = K; K = K+INC; IF... ; K = OLD.  IT COULDN'T RECOGNIZE  CCG0128    2    A
HISTORY            THE DIFFERENCE BETWEEN THE REDEFINITION (K=OLD) AND A        CCG0128    3    A
HISTORY            POST STORE FROM ANOTHER LOOP.                                CCG0128    4    A
HISTORY            DEPENDENCY=FCC0414,FCC0514,FCC1568X                          CCG0128    5    A
HISTORY            SIJ    84/06/14    APPLIES TO OPT=2                          CCG0128    6    A
HISTORY                                                                         CCG0128    7    A
HISTORY   CCG0146  GRA/JPX - FIX REGRESSION IN CCG0089.  RVT MUST BE SET        CCG0146    1    A
HISTORY            TO ZERO WHEN LEAVING LOOP.                                   CCG0146    2    A
HISTORY            DEPENDENCY=CCG0089                                           CCG0146    3    A
HISTORY            SIJ    84/06/05    GRA                                       CCG0146    4    A
HISTORY                                                                         CCG0146    5    A
HISTORY   CCG0147  SQZ/RJ3.  CORRECTLY PROCESS FORMAL PARAMETERS.               CCG0147    1    A
HISTORY            DEPENDENCY=CCGA083                                           CCG0147    2    A
HISTORY            SIJ    84/06/04    SQZ                                       CCG0147    3    A
HISTORY                                                                         CCG0147    4    A
HISTORY   CCG0150  SQZ/ARF.  ITERATE AS NECESSARY TO GET PAST REPEATED          CCG0150    1    A
HISTORY            CONSTANT INCREMENTS.                                         CCG0150    2    A
HISTORY            SIJ    84/06/22    APPLIES TO FTN5, OPT = 1,2                CCG0150    3    A
HISTORY                                                                         CCG0150    4    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN HISTORY

```
HISTORY   CCG0154  SQZ/CTI.  FIX LEVEL 2 AND FORMAL PARAM CONFLICT IN OPT=2.      CCG0154    1    A
HISTORY            SQZ DELETED THE XMT, AND GPO MADE A LOCAL COPY, SO X = FP      CCG0154    2    A
HISTORY            WAS COMPILED AS A LOAD FOLLOWED BY A STORE.                    CCG0154    3    A
HISTORY            SIJ    84/06/12    SQZ                                         CCG0154    4    A
HISTORY                                                                           CCG0154    5    A
HISTORY   *L642*    ************************************************             *L642*     1    A
HISTORY   CCGA090  COMQSVR/RSR, GPO/PCC.  VOID INSTRUCTION STACK AFTER PLUGGING   CCGA090    1    A
HISTORY            INSTRUCTIONS IN THE FOLLOWING PLACES -                         CCGA090    2    A
HISTORY            RSR IN COMQSVR, PCC5 IN PCC IN GPO.                            CCGA090    3    A
HISTORY            AXM    85/03/25    COMQSVR, GPO                                CCGA090    4    A
HISTORY                                                                           CCGA090    5    A
HISTORY   CCG0157  MCG/JAM/SNI.  ADDED CODE TO INSURE XSW IS NOT A -0.            CCG0157    1    A
HISTORY            SIJ    84/11/21    MCG                                         CCG0157    2    A
HISTORY                                                                           CCG0157    3    A
HISTORY   CCG0158  SQZ/MMR.  RJ3 - INVARIANT FUNCTION CALLS - FOLLOW MEMORY       CCG0158    1    A
HISTORY            REFERENCE CHAIN TO DEFINITION OF FIRST MEMORY REFERENCE.       CCG0158    2    A
HISTORY            DEPENDENCY=CCGA083                                             CCG0158    3    A
HISTORY            SIJ    84/10/23    SQZ                                         CCG0158    4    A
HISTORY                                                                           CCG0158    5    A
HISTORY   CCG0165  SQZ/RJ3.  REMOVE TWO SPURIOUS INSTRUCTIONS FROM *CCGA083*      CCG0165    1    A
HISTORY            WHICH CAUSED INCORRECT ANSWERS AT OPT=1,2 IN CERTAIN CASES     CCG0165    2    A
HISTORY            WHEN COMMON IS USED.                                           CCG0165    3    A
HISTORY            DEPENDENCY=CCGA083                                             CCG0165    4    A
HISTORY            SIJ    84/11/21    SQZ                                         CCG0165    5    A
HISTORY                                                                           CCG0165    6    A
HISTORY   *L650*    ************************************************             *L650*     1    A
HISTORY   CCG0172  GRA/IRA. IF MINUS IN B SWITCH RJ TO RK. THERE ARE NO           CCG0172    1    A
HISTORY            INSERT OR DELETE REFERENCES TO CCG0034 BUT THAT PSR            CCG0172    2    A
HISTORY            MUST BE INSTALLED FOR CCG0172 TO WORK PROPERLY.                CCG0172    3    A
HISTORY            DEPENDENCY=CCG0034                                             CCG0172    4    A
HISTORY            SSK    85/05/07    GRA                                         CCG0172    5    A
HISTORY                                                                           CCG0172    6    A
HISTORY   CCG0181  SQZ/RJ3.  AVOID UPDATING A REFERENCE TO ST.+C IF ITS           CCG0181    1    A
HISTORY            DESCRIPTOR IS ZERO.                                            CCG0181    2    A
HISTORY            DEPENDENCY=CCGA083                                             CCG0181    3    A
HISTORY            SSK    85/06/13    SQZ                                         CCG0181    4    A
HISTORY                                                                           CCG0181    5    A
HISTORY   *L670*    ************************************************             *L670*     1    A
HISTORY   CCGA091  GPO/UPB.  IF MOD LIST TABLE EMPTY, CHECK NT(NUM OF TEMP LDS)   CCGA091    1    A
HISTORY            FOR LOOP EXIT.                                                 CCGA091    2    A
HISTORY            DEPENDENCY=FCC2149                                             CCGA091    3    A
HISTORY            SSK    85/06/25    GPO                                         CCGA091    4    A
HISTORY                                                                           CCGA091    5    A
HISTORY   CCGA092  CGTM/F.RDT.  THIS PSR CONTAINS INSTRUCTION TIMINGS FOR THE     CCGA092    1    A
HISTORY            990 AND MUST BE INSTALLED CONCURRENTLY WITH FC5A495.           CCGA092    2    A
HISTORY            SSK    86/04/15    CGTM                                        CCGA092    3    A
HISTORY                                                                           CCGA092    4    A
HISTORY   CCGA093  MCG/SII.  INTRODUCE THE INTRINSIC FUNCTION *SUM1S* TO UTILIZE  CCGA093    1    A
HISTORY            THE POPULATION COUNT INSTRUCTION.  REQUIRES CONCURRENT         CCGA093    2    A
HISTORY            INSTALLATION OF FC5A496.                                       CCGA093    3    A
HISTORY            DEPENDENCY=CCG16                                               CCGA093    4    A
HISTORY            ASD/SSK 86/02/20   MCG                                         CCGA093    5    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN HISTORY

```
HISTORY                                                                    CCGA093     6    A
HISTORY   CCG0187  CGTM/CG$IEP.  ADDED A RECALL PARAMETER WHEN CLOSING F.OPT.  CCG0187     1    A
HISTORY            WITHOUT THE RECALL BIT, ANY DELAY BY THE OPERATING SYSTEM  CCG0187     2    A
HISTORY            IN COMPLETING THE CLOSE REQUEST CAN RESULT IN OVERLAY 23   CCG0187     3    A
HISTORY            BEING IN CORE BEFORE THE REQUEST IS COMPLETED RESULTING IN CCG0187     4    A
HISTORY            A WORD OF EXECUTABLE CODE IN PROGRAM MAP GETTING WIPED OUT. CCG0187     5    A
HISTORY            DF/SSK 86/06/19    CGTM                                    CCG0187     6    A
HISTORY                                                                    CCG0187     7    A
HISTORY   *L688*    ************************************************       *L688*      1    A
HISTORY   CCGA094  SQZ/RJ3.  REPAIR REGISTER WIPEOUT(A5), THE POINTER TO THE CCGA094     1    A
HISTORY            RJ3(BEF) CHAIN.                                         CCGA094     2    A
HISTORY            SSK    86/08/07    SQZ                                  CCGA094     3    A
HISTORY                                                                    CCGA094     4    A
HISTORY   CCG0177  BDT/RNI.  SOME DWL/DRL INSTRUCTIONS MAY CONTAIN IH.RF'S THAT CCG0177     1    A
HISTORY            DID NOT APPEAR AS AN RI WHICH DEFINES A RESULT IN A PREVIOUS CCG0177     2    A
HISTORY            INSTRUCTION WITHIN THE CURRENT INSTRUCTION SEQUENCE. AS A CCG0177     3    A
HISTORY            RESULT AN RF MAY HAVE AN R NUMBER DEFINITION TABLE FORMAT CCG0177     4    A
HISTORY            WORD ADDED RESULTING IN A MODEOUT LATER. THIS CODE PREVENTS CCG0177     5    A
HISTORY            RELABELING OF AN RF IF NOTHING WAS SETUP FOR THIS RF IN THE CCG0177     6    A
HISTORY            EQUIVALENCE TABLE.                                      CCG0177     7    A
HISTORY            DEPENDENCY=FCC0643                                      CCG0177     8    A
HISTORY            SSK    87/01/26    BDT                                  CCG0177     9    A
HISTORY                                                                    CCG0177    10    A
HISTORY   CCG0188  BDT/FTL. THE *BMC* LIST RUNS INTO THE TEMPORARY TREE FOR THIS CCG0188     1    A
HISTORY            TEST CASE CAUSING BDT TO MODE OUT WHILE FORMING BOUNDARY CCG0188     2    A
HISTORY            MARKER LINKS.                                           CCG0188     3    A
HISTORY            SSK    87/03/10    BDT                                  CCG0188     4    A
HISTORY                                                                    CCG0188     5    A
HISTORY   CCG0189  SQZ/MMR.  SEARCH MEMORY REFERENCE CHAIN BACK TO DEFINITION CCG0189     1    A
HISTORY            OF FIRST MEMORY REFERENCE FOR BEF'S.                    CCG0189     2    A
HISTORY            DEPENDENCY=CCGA083                                      CCG0189     3    A
HISTORY            SSK    86/08/01    SQZ                                  CCG0189     4    A
HISTORY                                                                    CCG0189     5    A
HISTORY   CCG0191  MCG/CRA/JAM. IF MOVING *B* TO AN *X*, ENSURE *X* IS NOT IN K CCG0191     1    A
HISTORY            FIELD OF AN *SA* INSTRUCTION TO BE ISSUED. IF SO SWITCH J CCG0191     2    A
HISTORY            AND K.                                                  CCG0191     3    A
HISTORY            DEPENDENCY=FCC1555X                                     CCG0191     4    A
HISTORY            SSK    87/01/09    MCG                                  CCG0191     5    A
HISTORY                                                                    CCG0191     6    A
HISTORY   CCG0193  SQZ/RJ3.  IF THE CURRENT DEF WAS MARKED DEAD THEN MARK THE CCG0193     1    A
HISTORY            CURRENT RJ3 AS NOT NECESSARY.                           CCG0193     2    A
HISTORY            DEPENDENCY=CCGA083                                      CCG0193     3    A
HISTORY            SSK    86/09/22    SQZ                                  CCG0193     4    A
HISTORY                                                                    CCG0193     5    A
HISTORY   CCG0196  MCG/SNI.  ADDED CODE TO INSURE *XSW* IS NOT A -0.  PREVENTS CCG0196     1    A
HISTORY            ENDLESS LOOP IN *MCG*.                                  CCG0196     2    A
HISTORY            SSK    86/11/12    MCG                                  CCG0196     3    A
HISTORY                                                                    CCG0196     4    A
HISTORY   *L716*    ************************************************       *L716*      1    A
HISTORY   CCG0206  SQZ/RJ3.  DO NOT ELIMINATE CURRENT RJ3 AND FORCE DEPENDENCY CCG0206     1    A
HISTORY            ON AN RJ3 WITH NO FOLLOWING OC.DEF.                     CCG0206     2    A
HISTORY            DEPENDENCY=CCGA083                                      CCG0206     3    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN HISTORY

```
HISTORY            SSK    88/01/22    SQZ                                          CCG0206        4     A
HISTORY                                                                            CCG0206        5     A
HISTORY   CCG0207  BDT/CIP.  USE JAM MODE PRIORITY CALCULATION WHEN A CODE          CCG0207        1     A
HISTORY            SEQUENCE HAS A *LARGE* NUMBER OF STORE INSTRUCTIONS.             CCG0207        2     A
HISTORY            SSK    87/10/22    BDT                                          CCG0207        3     A
HISTORY                                                                            CCG0207        4     A
HISTORY   CCG0208  GPO/RDD.  IF AN INSTRUCTION IS AN IMMEDIATE PREDECESSOR OF       CCG0208        1     A
HISTORY            A LEVEL 0 STORE, THEN SET THE *PS* FLAG.                         CCG0208        2     A
HISTORY            SSK    87/11/11    GPO                                          CCG0208        3     A
HISTORY                                                                            CCG0208        4     A
HISTORY   CCG0211  SQZ/RJ3.  SET UP THE UDI LINK IN THE LD, IF IT IS NECESSARY      CCG0211        1     A
HISTORY            TO GENERATE A LD/XMT.                                            CCG0211        2     A
HISTORY            DEPENDENCY=CCGA083                                               CCG0211        3     A
HISTORY            SSK    88/01/22    SQZ                                          CCG0211        4     A
HISTORY                                                                            CCG0211        5     A
HISTORY   CCG0212  GPO/UPB.  INHIBIT STT'S IF LCM=G AND OPT=2.                      CCG0212        1     A
HISTORY            DEPENDENCY=CCG21,FC50717                                         CCG0212        2     A
HISTORY            SSK    88/02/03    GPO                                          CCG0212        3     A
HISTORY                                                                            CCG0212        4     A
HISTORY   CCG0213  SQZ/IXI.  AVOID ADJUSTING RF-S FOR TYPE FOUR INSTRUCTIONS.       CCG0213        1     A
HISTORY            DEPENDENCY=CCG17                                                 CCG0213        2     A
HISTORY            SSK    88/02/12    SQZ                                          CCG0213        3     A
HISTORY                                                                            CCG0213        4     A
HISTORY   *L739*    ***********************************************               *L739*         1     A
HISTORY   CCG0214  GPO/UPB.  THE TEST CASE SUBMITTED FOR THIS PSR CAUSES THE        CCG0214        1     A
HISTORY            CODE GENERATOR TO HANG. CAUSED BY CCG0212 WHICH WIPES OUT        CCG0214        2     A
HISTORY            A REGISTER. THIS PSR CORRECTS THE REGISTER WIPEOUT.              CCG0214        3     A
HISTORY            DEPENDENCY=FCC0717                                               CCG0214        4     A
HISTORY            SSK    88/06/01    GPO                                          CCG0214        5     A
HISTORY                                                                            CCG0214        6     A
HISTORY   CCG0215  BDT/SIO.  WHEN MOVING INSTRUCTIONS TO THEIR FINAL ORDER, IF      CCG0215        1     A
HISTORY            AN INSTRUCTION IS NOT *MARKED*, USE THE RI OF THE INSTRUCTION    CCG0215        2     A
HISTORY            AS THE OUTPUT INDEX(I.OI).                                       CCG0215        3     A
HISTORY            ***THIS PSR REQUIRES CONCURRENT INSTALLATION OF CCG0216.***      CCG0215        4     A
HISTORY            SSK    88/06/16    BDT                                          CCG0215        5     A
HISTORY                                                                            CCG0215        6     A
HISTORY   CCG0216  SQZ/RJ3.  WHEN CHANGING AN RJ3/DEF TO LD/XMT AND DOUBLE          CCG0216        1     A
HISTORY            PRECISION IS INVOLVED, CHECK TO SEE IF THE MRC CHAIN IS          CCG0216        2     A
HISTORY            POINTING TO THE SECOND GTL, IF IT IS, THE MRC CHAIN WILL         CCG0216        3     A
HISTORY            BE UPDATED.                                                      CCG0216        4     A
HISTORY            DEPENDENCY=CCGA083                                               CCG0216        5     A
HISTORY            SSK    88/06/16    SQZ                                          CCG0216        6     A
HISTORY                                                                            CCG0216        7     A
HISTORY   CCG0218  GPO/MII.  CORRECTIVE ADDED TO CHECK FOR THE SECOND *DEF*         CCG0218        1     A
HISTORY            IN CASE AN *RJ3* IS RETURNING A DOUBLE PRECISION RESULT.         CCG0218        2     A
HISTORY            SSK    88/08/04    GPO                                          CCG0218        3     A
HISTORY                                                                            CCG0218        4     A
HISTORY   CCG0220  CGTM/CG$CUB.  WHEN CHANGING USE BLOCKS, THE PARCEL COUNT IS      CCG0220        1     A
HISTORY            SAVED IN THE WRONG FIELD WITH THE CURRENT BLOCK LENGTH.          CCG0220        2     A
HISTORY            WHEN THE REAR-END-CONTROLLER(REC) PROCESSES THE LOCAL BLOCK      CCG0220        3     A
HISTORY            TABLES IT NEEDS THE PARCEL COUNT FOR A TABLE TO ROUND UP IF      CCG0220        4     A
HISTORY            NECESSARY. ADDED CORRECTIVE CODE FIXES PROBLEM.                  CCG0220        5     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN HISTORY

```
HISTORY            DEPENDENCY=CCG15                                          CCG0220        6     A
HISTORY            SSK    89/03/03    CGTM                                   CCG0220        7     A
HISTORY                                                                      CCG0220        8     A
HISTORY   *L780*    ************************************************         *L780*         1     A
HISTORY   CCGA097  GRA/IRA.  AN IA INSTRUCTION WITH ONE OF THE OPERANDS IN A B  CCGA097      1     A
HISTORY            REGISTER CAN'T BE CHANGED TO AN SA BECAUSE THE RESULT IS NOT  CCGA097      2     A
HISTORY            SHORT USAGE(LCM=G). CORRECTIVE CODE INSERTS AN SA TO AN X  CCGA097        3     A
HISTORY            REGISTER BEFORE THE IA.                                   CCGA097        4     A
HISTORY            DEPENDENCY= CCG0034                                       CCGA097        5     A
HISTORY            SSK    89/12/15    GRA                                    CCGA097        6     A
HISTORY                                                                      CCGA097        7     A
HISTORY   *L797*    ************************************************         *L797*         1     A
HISTORY   CCGA099  GRA/SCT.  WHEN SELECTING CANDIDATES FOR B REGISTER ADDRESSING IN  CCGA099  1     A
HISTORY            LOOP OPTIMIZATION, DISALLOW LEVEL 0 ARGUMENTS WHEN LCM=I OR LCM=G.  CCGA099  2     A
HISTORY            PROSEQ/ERS.  EXPAND SPECIAL REFERENCES IN PROSEQ IS ALLOWING  CCGA099      3     A
HISTORY            A B REGISTER ASSIGNMENT TO A LEVEL 0 FORMAL PARAMETER.    CCGA099        4     A
HISTORY            CORRECTIVE CODE CHANGES AN LCM=G TEST TO A NOT LCM=D TEST.  CCGA099       5     A
HISTORY            DEPENDENCY=CCGA044                                        CCGA099        6     A
HISTORY            DEPENDENCY=FCC0594                                        CCGA099        7     A
HISTORY            PID=FC5                                                   CCGA099        8     A
HISTORY            MRR    92/02/04    GRA, PROSEQ                            CCGA099        9     A
HISTORY                                                                      CCGA099       10     A
HISTORY   CCG1*92  PROBLEM  - CONTROL DATA CORPORATION HAS CHANGED ITS NAME  CCG1*92         1     A
HISTORY                       TO CONTROL DATA SYSTEMS INC, SO ALL COPYRIGHT  CCG1*92         2     A
HISTORY                       INFORMATION MUST BE UPDATED.                   CCG1*92         3     A
HISTORY                                                                      CCG1*92         4     A
HISTORY            SOLUTION - UPDATE COPYRIGHT INFORMATION.                  CCG1*92         5     A
HISTORY                                                                      CCG1*92         6     A
HISTORY            BAO    09/15/92                                           CCG1*92         7     A
HISTORY                                                                      CCG1*92         8     A
HISTORY   *L803*    ************************************************         *L803*         1     A
HISTORY   CCGA100  PROBLEM  - CCGA099 CORRECTIVE CODE WAS IRRELEVANT.        CCGA100        1     A
HISTORY                       THE B REGISTER WAS INCORRECTLY ASSIGNED IN ROUTINE  CCGA100    2     A
HISTORY                       MFA (MAKE FINAL B-ASSIGNMENTS).                CCGA100        3     A
HISTORY            SOLUTION - REMOVE THE CCGA099 CODE FROM ROUTINE SCT.      CCGA100        4     A
HISTORY                       ADD CODE TO MFA TO RECOGNIZE AN LCM VARIABLE AND  CCGA100      5     A
HISTORY                       AVOID ASSIGNING A -B- REGISTER FOR IT.         CCGA100        6     A
HISTORY            RLM     93/07/01   GRA, PROSEQ                           CCGA100        7     A
HISTORY                                                                      CCGA100        8     A
HISTORY   *L826*    ************************************************         *L826*         1     A
HISTORY   CCG*94   PROBLEM -  COPYRIGHT STATEMENTS NEED TO BE UPDATED.       CCG*94         1     A
HISTORY                                                                      CCG*94         2     A
HISTORY            SOLUTION - UPDATE ALL COPYRIGHTS TO 1994 AND CHANGE ANY REFERENCES  CCG*94  3     A
HISTORY                       TO CONTROL DATA CORPORATION TO CONTROL DATA SYSTEMS INC.  CCG*94  4     A
HISTORY                                                                      CCG*94         5     A
HISTORY            BAO    04/19/94    CCOMISC,CCOMLOL,COMAREG,COMCLFM,COMCSBM,CGTM  CCG*94  6     A
HISTORY   *L840*    ************************************************         *L840*         1     A
HISTORY   *L847*    ************************************************         *L847*         1     A
HISTORY   *L851*    ************************************************         *L851*         1     A
HISTORY   *L859*    ************************************************         *L859*         1     A
HISTORY   *L871*    ************************************************         *L871*         1     A
HISTORY                                                                      HISTORY        2     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - HISTORY

IDENTIFIER	TOTAL	ACTIVE
HISTORY	2	2
FCC0885XA	4	4
FCC1655X	3	3
FCC1656X	4	4
FCC1660X	5	5
FCC1584X	4	4
FCC1677X	4	4
FCC1695X	4	4
FCC1700X	3	3
FCC1710X	5	5
FCC1719X	4	4
FCC1739X	4	4
FCC1740X	4	4
FCC1755X	4	4
FCC1770X	5	5
FCC1825X	4	4
FCC1828X	5	5
FCC1837X	5	5
FCC1845X	4	4
FCC1858X	3	3
FCC0238AX	5	5
CCG*78	3	3
FCC1908	4	4
FCC1918	6	6
FCC1928	5	5
CCGA005	4	4
CCGA006	9	9
CCGA007	3	3
FCC1997	4	4
CCG25	5	5
CCG26	6	6
CCG27	3	3
FCC2038	4	4
FCC2045	3	3
FCC2056	5	5
FCC2051	4	4
CCGA008	3	3
FCC2072	6	6
CCG31	1	1
CCG36	4	4
CCG32	5	5
CCG34	4	4
CCG35	3	3
FCC2074	4	4
FCC2091	5	5
FCCA462	4	4
CCG38	4	4
CCG39	4	4
CCG40	4	4
L498	1	1

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN HISTORY

CCGA009	4	4
L505	1	1
F2950CC	4	4
CCGA010	3	3
CCGA011	3	3
CCGA012	3	3
CCGA013	4	4
CCGA014	3	3
CCGA015	3	3
L508	1	1
FCCA499X	4	4
FCCA506X	3	3
FCC2212X	6	6
CCGA017	5	5
CCGA018	4	4
CCGA019	9	9
CCGA021	7	7
CCGA022	4	4
CCGA024	4	4
L518	1	1
CCG*80	3	3
CCGA023	10	10
CCGA025	4	4
CCGA026	5	5
FCCA509X	6	6
FCCA513X	7	7
FCC2230X	6	6
FCC2271X	4	4
FCC2273X	8	8
L528	1	1
F1037CC	4	4
CCGA027	5	5
CCGA028	4	4
CCGA029	4	4
CCGA030	4	4
CCGA031	8	8
CCGA034	4	4
FCCA546X	6	6
FCCA550X	4	4
L538	1	1
CCG*81	3	3
CCGA035	13	13
CCGA036	17	17
CCGA037	6	6
CCGA039	4	4
L552	1	1
CCGA040	4	4
CCGA041	5	5
CCGA042	5	5
CCGA043	5	5
CCGA044	8	8
CCGA045	13	13

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN HISTORY

CCGA046	11	11
CCGA047	5	5
CCG0001	4	4
CCG0004	6	6
L564	1	1
CCG*82	3	3
CCGA049	11	11
CCGA052	8	8
CCGA054	4	4
CCG0006	4	4
CCG0013	5	5
L577	1	1
CCGA062	4	4
CCGA063	5	5
CCGA066	4	4
CCGA067	4	4
CCGA068	10	10
CCGA072	5	5
CCGA073	7	7
CCG0002	4	4
CCG0008	5	5
CCG0018	4	4
CCG0022	4	4
L587	1	1
CCGA076	6	6
CCGA079	5	5
CCGA082	5	5
CCG0037	4	4
CCG0041	8	8
CCG0045	5	5
CCG0071	6	6
CCG0085	4	4
L601	1	1
CCGA083	5	5
CCG0012	7	7
CCG0034	5	5
CCG0057	5	5
CCG0086	10	10
CCG0088	13	13
CCG0089	5	5
L617	1	1
CCG0093	4	4
CCG0110	6	6
CCG0117	5	5
CCG0119	5	5
CCG0124	5	5
L628	1	1
CCGA085	5	5
CCGA088	5	5
CCG0127	7	7
CCG0128	7	7
CCG0146	5	5

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN HISTORY

CCG0147	4	4
CCG0150	4	4
CCG0154	5	5
L642	1	1
CCGA090	5	5
CCG0157	3	3
CCG0158	5	5
CCG0165	6	6
L650	1	1
CCG0172	6	6
CCG0181	5	5
L670	1	1
CCGA091	5	5
CCGA092	4	4
CCGA093	6	6
CCG0187	7	7
L688	1	1
CCGA094	4	4
CCG0177	10	10
CCG0188	5	5
CCG0189	5	5
CCG0191	6	6
CCG0193	5	5
CCG0196	4	4
L716	1	1
CCG0206	5	5
CCG0207	4	4
CCG0208	4	4
CCG0211	5	5
CCG0212	4	4
CCG0213	4	4
L739	1	1
CCG0214	6	6
CCG0215	6	6
CCG0216	7	7
CCG0218	4	4
CCG0220	8	8
L780	1	1
CCGA097	7	7
L797	1	1
CCGA099	10	10
CCG1*92	8	8
L803	1	1
CCGA100	8	8
L826	1	1
CCG*94	6	6
L840	1	1
L847	1	1
L851	1	1
L859	1	1
L871	1	1

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN HISTORY


          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN *COMDKS*

*COMDKS* *DECK  *COMDKS*                                                    *COMDKS*       1     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - *COMDKS*

          IDENTIFIER                TOTAL    ACTIVE

          *COMDKS*                      1         1

          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCGTMTV

CCGTMTV  *COMDECK  CCGTMTV            TABLE MANAGER TABLE VECTOR            CCGTMTV        1     A
CCGTMTV                                                                     CCGTMTV        2     A
CCGTMTV  **        CCGTMTV - CCG TABLE MANAGER TABLE VECTOR                 CCGTMTV        3     A
CCGTMTV  *         THIS COMDECK CONSISTS OF A SEQUENCE OF CALLS TO THE *TABLE*  CCGTMTV    4     A
CCGTMTV  *         MACRO WHICH DEFINES THE ORDER AND NUMBER OF TABLES THAT ARE  CCGTMTV    5     A
CCGTMTV  *         TO BE MANAGED DURING CCG PROCESSING.                     CCGTMTV        6     A
CCGTMTV  *                                                                  CCGTMTV        7     A
CCGTMTV  *         THE FORMAT OF THE MACRO CALL IS -                        CCGTMTV        8     A
CCGTMTV  * TNAM    TABLE  EQUIV                                             CCGTMTV        9     A
CCGTMTV  *         WHERE -                                                  CCGTMTV       10     A
CCGTMTV  *         TNAM - TABLE NAME                                        CCGTMTV       11     A
CCGTMTV  *         EQUIV - EQUIVALENT TABLE NAME, IF PRESENT.  THIS ALLOWS  CCGTMTV       12     A
CCGTMTV  *         CERTAIN TABLES TO BE USED BY DIFFERENT PROCESSORS.       CCGTMTV       13     A
CCGTMTV                                                                     CCGTMTV       14     A
CCGTMTV  *         THE COLLECTION OF TABLE MACRO CALLS SHOULD GENERATE THE  CCGTMTV       15     A
CCGTMTV  *         FOLLOWING QUANTITIES -                                   CCGTMTV       16     A
CCGTMTV  *         A) A VECTOR OF CONTIGIOUS CELLS HOLDING THE FIRST WORD ADDRESS  CCGTMTV 17     A
CCGTMTV  *         OF EACH TABLE. THE ENTRY POINT NAME OF THE TABLE FWA IS O$TNAM  CCGTMTV 18     A
CCGTMTV  *         B) A VECTOR OF CONTIGIOUS CELLS HOLDING THE LENGTH OF EACH  CCGTMTV    19     A
CCGTMTV  *         TABLE.  THE ENTRY POINT NAME OF THE TABLE LENGTH CELL IS  CCGTMTV       20     A
CCGTMTV  *         L$TNAM .                                                 CCGTMTV       21     A
CCGTMTV  *         C) FOR EACH TABLE, ITS INDEX ( Z$TNAM ) INTO THE ABOVE VECTORS.  CCGTMTV 22     A
CCGTMTV                                                                     CCGTMTV       23     A
CCGTMTV  **        B$TBL - BASIC TABLE MANAGER TABLE VECTOR DEFINITION MACRO  CCGTMTV       24     A
CCGTMTV  *         DEFINES O$TNAM, L$TNAM AND Z$TNAM AS ENTRY POINTS        CCGTMTV       25     A
CCGTMTV                                                                     CCGTMTV       26     A
CCGTMTV            PURGMAC B$TBL                                            CCGTMTV       27     A
CCGTMTV            MACRO  B$TBL,TNAM,EQUIV                                  CCGTMTV       28     A
CCGTMTV            ENTRY  O$TNAM,L$TNAM,Z$TNAM                              CCGTMTV       29     A
CCGTMTV   T        IFC    EQ,/EQUIV//                                       CCGTMTV       30     A
CCGTMTV   Z$TNAM   EQU    *-FTAB                                            CCGTMTV       31     A
CCGTMTV   O$TNAM   CON    0                                                 CCGTMTV       32     A
CCGTMTV   TABLES   RMT                                                      CCGTMTV       33     A
CCGTMTV            ORG    LTAB+Z$TNAM                                       CCGTMTV       34     A
CCGTMTV   L$TNAM   CON    0                                                 CCGTMTV       35     A
CCGTMTV   TABLES   RMT                                                      CCGTMTV       36     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCGTMTV

```
CCGTMTV   T        ELSE                                                    CCGTMTV      37     A
CCGTMTV   Z$TNAM   EQU    Z$EQUIV                                          CCGTMTV      38     A
CCGTMTV   O$TNAM   EQU    O$EQUIV                                          CCGTMTV      39     A
CCGTMTV   TABLES   RMT                                                     CCGTMTV      40     A
CCGTMTV   L$TNAM   EQU    L$EQUIV                                          CCGTMTV      41     A
CCGTMTV   TABLES   RMT                                                     CCGTMTV      42     A
CCGTMTV   T        ENDIF                                                   CCGTMTV      43     A
CCGTMTV   B$TBL    ENDM                                                    CCGTMTV      44     A
CCGTMTV                                                                    CCGTMTV      45     A
CCGTMTV   BLK      TABLE             BLOCKS IN CORE ( OPT=2 )              CCGTMTV      46     A
CCGTMTV                                                                    CCGTMTV      47     A
CCGTMTV   TXT      TABLE             CURRENT IL INSTRUCTION SEQUENCE       CCGTMTV      48     A
CCGTMTV   GST      TABLE  TXT        GRAPH STRUCTURE TABLE ( CFA )         CCGTMTV      49     A
CCGTMTV                                                                    CCGTMTV      50     A
CCGTMTV   RND      TABLE             R-NUMBER DEFINITION S ( SQZ )         CCGTMTV      51     A
CCGTMTV   DUT      TABLE  RND        DEF-USE           ( PROSEQ/AIS )      CCGTMTV      52     A
CCGTMTV   TREE     TABLE  RND        DEPENDENCY TREE/SUCC INDEX TBL ( BDT/MCG )  CCGTMTV  53     A
CCGTMTV                                                                    CCGTMTV      54     A
CCGTMTV   PIT      TABLE             POSTED INSTRUCTIONS ( MCG/WII )       CCGTMTV      55     A
CCGTMTV   PBT      TABLE  PIT        PRE ASSIGNED B-REGS ( PROSEQ/AIS )    CCGTMTV      56     A
CCGTMTV                                                                    CCGTMTV      57     A
CCGTMTV   OTI      TABLE             OT. INFO 12/P(OT. ORD),30/,18/RI IN OT.  CCGTMTV   58     A
CCGTMTV   MLT      TABLE  OTI        MOD LIST INDEX TABLE ( WHEN CALLING MPB )  CCGTMTV 59     A
CCGTMTV                                                                    CCGTMTV      60     A
CCGTMTV   MOD      TABLE             MODIFICATIONS TO BE INSERT/DELETED ( MPB )  CCGTMTV 61    A
CCGTMTV                                                                    CCGA011       4     A
CCGTMTV  ******CUTOFF POINT OF HIGHLY ACTIVE TABLES FOR OPT 2, PHASE 2 *********  CCGA011  5  A
CCGTMTV                                                                    CCGA011       6     A
CCGTMTV   RCT      TABLE                                                   CCGA011       7     A
CCGTMTV   BCT      TABLE  RCT                                              CCGA011       8     A
CCGTMTV                                                                    CCGA011       9     A
CCGTMTV  ************CUTOFF POINT OF HIGHLY ACTIVE TABLES FOR OPT 1*************  CCGA011  10  A
CCGTMTV                                                                    CCGTMTV      62     A
CCGTMTV   CFT      TABLE             CONTROL FLOW TABLE ( BRIDGE AND CFA )  CCGTMTV      63     A
CCGTMTV                                                                    CCGA011      11     A
CCGTMTV  ******CUTOFF POINT OF HIGHLY ACTIVE TABLES FOR OPT 2, PHASE 1 *********  CCGA011  12  A
CCGTMTV                                                                    CCGTMTV      64     A
CCGTMTV   BST      TABLE             BLOCK STATUS INFORMATION              CCGTMTV      65     A
CCGTMTV  * BIT     #      BST        ( PHASE 1 OF OPT=2 )                  CCGTMTV      66     A
CCGTMTV                                                                    CCGTMTV      67     A
CCGTMTV   PSI      TABLE             POST STORE INFORMATION ( GPO/GRA )    CCGTMTV      68     A
CCGTMTV                                                                    CCGTMTV      69     A
CCGTMTV   RXI      TABLE             REGION EXIT INFORMATION ( GPO/GRA )   CCGTMTV      70     A
CCGTMTV                                                                    CCGTMTV      71           I
          -CCGA011
CCGTMTV   RCT      TABLE             REGISTER CANIDATE TABLE ( GRA )       CCGTMTV      72           I
          -CCGA011
CCGTMTV   BCT      TABLE  RCT        B-CANIDATE TABLE  ( PROSEQ/AIS )     CCGTMTV      73           I
          -CCGA011
CCGTMTV                                                                    CCGTMTV      74     A
CCGTMTV   TET      TABLE             TEMP EQUIVALENCE TABLE ( CODE MOTION )  CCGTMTV      75     A
CCGTMTV                                                                    CCGTMTV      76     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCGTMTV

CCGTMTV   IIT      TABLE              INCREMENT INFORMATION ( GPO/GRA )                CCGTMTV       77     A
CCGTMTV   IOL      TABLE  IIT         I/O APLIST PARAMETER ORDINALS ( OPT=2 )          CCG18          1     A
CCGTMTV                                                                                CCGTMTV       78     A
CCGTMTV   UDI      TABLE              USE/DEF INDEX TABLE ( BRIDGE/UDT )               CCGTMTV       79     A
CCGTMTV                                                                                CCGTMTV       80     A
CCGTMTV  *         CUT AND CVT MUST BE INCLUDED IN *CGHCDTD*                           CCGTMTV       81     A
CCGTMTV  *         SYM, CBT AND THE FOWARD REFERENCE ADDRESS DEFINITION TABLES         CCGTMTV       82     A
CCGTMTV  *         MAY BE IN EITHER OF THE BELOW COMDECKS.                             CCGTMTV       83     A
CCGTMTV                                                                                CCGTMTV       84     A
CCGTMTV  *CALL     CGHCDTD            HOST COMPILER DYNAMIC TABLE DECLARATIONS         CCGTMTV       85     A
CCGTMTV                                                                                CCGTMTV       86     A
CCGTMTV            IF     DEF,.T,1                                                     CCGTMTV       87     A
CCGTMTV   N.DT     EQU    .T          N. DYNAMIC TABLES                                CCGTMTV       88     A
CCGTMTV                                                                                CCGTMTV       89     A
CCGTMTV  *CALL     CGHCSTD            HOST COMPILER STATIC TABLES                      CCGTMTV       90     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - CCGTMTV

                IDENTIFIER                TOTAL    ACTIVE

                CCGTMTV                      90        87
                CCG18                         1         1
                CCGA011                       9         9

                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCOMGCM

CCOMGCM  *COMDECK  CCOMGCM            GENERAL COMPILER MACROS                          CCOMGCM        1     A
CCOMGCM            CTEXT  CCOMGCM - GENERAL COMPILER MACROS                            CCOMGCM        2     A
CCOMGCM   LXQ      SPACE  3,18                                                         CCOMGCM        3     A
CCOMGCM  **        LXQ - REDEFINE THE LEFT SHIFT INSTRUCTION.                          CCOMGCM        4     A
CCOMGCM  *                                                                             CCOMGCM        5     A
CCOMGCM  *         THIS OPDEF REDEFINES THE LEFT SHIFT INSTRUCTION TO SUPPRESS         CCOMGCM        6     A
CCOMGCM  *         CODE GENERATION WHEN THE SHIFT COUNT IS 0, +60D OR -60D.            CCOMGCM        7     A
CCOMGCM  *         THE INSTRUCTION IS OTHERWISE UNCHANGED.                             CCOMGCM        8     A
CCOMGCM  *                                                                             CCOMGCM        9     A
CCOMGCM  *         LXI       JK                                                        CCOMGCM       10     A
CCOMGCM  *                                                                             CCOMGCM       11     A
CCOMGCM  *         ENTRY  *XI* = X-REG TO BE SHIFTED                                   CCOMGCM       12     A
CCOMGCM  *                *JK* = SHIFT COUNT EXPRESSION                                CCOMGCM       13     A
CCOMGCM  *                                                                             CCOMGCM       14     A
CCOMGCM  *         USES   XI                                                           CCOMGCM       15     A
CCOMGCM                                                                                CCOMGCM       16     A
CCOMGCM   ^XQ      CPSYN     LXQ                                                       CCOMGCM       17          I
            -CCG26
CCOMGCM            PURGDEF   LXQ                                                       CCOMGCM       18          I
            -CCG26
CCOMGCM            PURGDEF  ^XQ                                                        CCG26          7     A
CCOMGCM            PURGDEF  LXQ                                                        CCG26          8     A
CCOMGCM   ^XQ      CPOP   0,200B,100B                                                  CCG26          9     A
CCOMGCM                                                                                CCOMGCM       19     A
CCOMGCM   LXQ      OPDEF     I,JK                                                      CCOMGCM       20     A
CCOMGCM    IFNE JK,0,2                                                                 CCOMGCM       21     A

            0         1         2         3         4         5         6         7         8
            12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCOMGCM

```
CCOMGCM    IFNE JK_&60D,0,1                                                    CCOMGCM      22    A
CCOMGCM    ^X.I JK                                                             CCOMGCM      23    A
CCOMGCM    ENDM                                                                CCOMGCM      24    A
CCOMGCM   RPVDEF   SPACE  4,8                                                  CCOMGCM      25    A
CCOMGCM  **        RPVDEF - DEFINE FWA OF ROUTINE FOR REPRIEVE UTILITY.        CCOMGCM      26    A
CCOMGCM  *                                                                     CCOMGCM      27    A
CCOMGCM  *         DEFINES THE FIRST WORD ADDRESS OF A ROUTINE AND DECLARES    CCOMGCM      28    A
CCOMGCM  *         IT AS AN ENTRY POINT, FOR FORMING THE NAME/ADDRESS TABLES   CCOMGCM      29    A
CCOMGCM  *         USED BY THE UTILITY ROUTINE *RPV* (LOCATED IN *LSTPRO*).    CCOMGCM      30          I
         -CCG26
CCOMGCM  *         USED BY THE UTILITY ROUTINE *RPV* (LOCATED IN THE CRADLE).  CCG26        10    A
CCOMGCM  *         FOR PROPER OPERATION, THIS MACRO MUST BE CALLED AT THE      CCOMGCM      31    A
CCOMGCM  *         BEGINNING OF EACH ROUTINE, BEFORE ANY OTHER INSTRUCTION OR  CCOMGCM      32    A
CCOMGCM  *         PSEUDO-OP THAT WOULD CAUSE *COMPASS-S* LOCATION COUNTER TO BE CCOMGCM    33    A
CCOMGCM  *         ADVANCED.                                                   CCOMGCM      34    A
CCOMGCM  *                                                                     CCOMGCM      35    A
CCOMGCM  *                                                                     CCOMGCM      36          I
         -CCG26
CCOMGCM  *  *TAG*  RPVDEF                                                      CCOMGCM      37          I
         -CCG26
CCOMGCM  *                                                                     CCOMGCM      38          I
         -CCG26
CCOMGCM  *         ENTRY  *TAG* = SYMBOLIC NAME OF ROUTINE FWA.                CCOMGCM      39          I
         -CCG26
CCOMGCM                                                                        CCOMGCM      40          I
         -CCG26
CCOMGCM                                                                        CCOMGCM      41          I
         -CCG26
CCOMGCM            PURGMAC   RPVDEF                                            CCOMGCM      42          I
         -CCG26
CCOMGCM                                                                        CCOMGCM      43          I
         -CCG26
CCOMGCM            MACRO  RPVDEF,T                                             CCOMGCM      44          I
         -CCG26
CCOMGCM            NOREF  T                                                    CCOMGCM      45          I
         -CCG26
CCOMGCM            ENTRY  T                                                    CCOMGCM      46          I
         -CCG26
CCOMGCM   T        BSS    0                                                    CCOMGCM      47          I
         -CCG26
CCOMGCM   RPVDEF   ENDM                                                        CCOMGCM      48          I
         -CCG26
CCOMGCM  * RNAM    RPVDEF ENAM                                                 CCG26        11    A
CCOMGCM  *                                                                     CCG26        12    A
CCOMGCM  *         ENTRY  RNAM = ROUTINE NAME                                  CCG26        13    A
CCOMGCM  *                ENAM = ENTRY POINT WILL BE B=*ENAM*.  IF ENAM IS     CCG26        14    A
CCOMGCM  *                       ABSENT, THE FIRST 5 CHARACTERS OF *RNAM* ARE  CCG26        15    A
CCOMGCM  *                       APPENDED TO *B=* ( IF NOT PRESENT ALREADY ).  CCG26        16    A
CCOMGCM                                                                        CCG26        17    A
CCOMGCM            PURGMAC RPVDEF                                              CCG26        18    A
CCOMGCM                                                                        CCG26        19    A
CCOMGCM            MACRO  RPVDEF,R,E                                           CCG26        20    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCOMGCM

```
CCOMGCM   '?RPV=A  MICRO  1,7, R                                          CCG26      21    A
CCOMGCM            IFC    NE,/E//,1                                       CCG26      22    A
CCOMGCM   '?RPV=A  MICRO  1,7, E                                          CCG26      23    A
CCOMGCM   '?RPV=B  MICRO  1,2, "'?RPV=A"                                  CCG26      24    A
CCOMGCM            IFC    NE,/"'?RPV=B"/B=/,1                             CCG26      25    A
CCOMGCM   '?RPV=A  MICRO  1,7, B="'?RPV=A"                                CCG26      26    A
CCOMGCM            NOREF  "'?RPV=A"                                       CCG26      27    A
CCOMGCM            ENTRY  "'?RPV=A"                                       CCG26      28    A
CCOMGCM   "'?RPV=A"  BSS  0                                               CCG26      29    A
CCOMGCM   RPVDEF   ENDM                                                   CCG26      30    A
CCOMGCM            SPACE  4,8                                             CCOMGCM    49          I
          -CCG26
CCOMGCM   **       FWA - DEFINE ENTRY FOR *RPV* FWA TABLE.                CCOMGCM    50          I
          -CCG26
CCOMGCM   *                                                               CCOMGCM    51          I
          -CCG26
CCOMGCM   *        FWA    NAM,EPN                                         CCOMGCM    52          I
          -CCG26
CCOMGCM   *                                                               CCOMGCM    53          I
          -CCG26
CCOMGCM   *        ENTRY  *NAM* = NAME OF SUBROUTINE                      CCOMGCM    54          I
          -CCG26
CCOMGCM   *               *EPN* = FWA NAME                                CCOMGCM    55          I
          -CCG26
CCOMGCM                                                                   CCOMGCM    56          I
          -CCG26
CCOMGCM            PURGMAC FWA                                            CCOMGCM    57          I
          -CCG26
CCOMGCM                                                                   CCOMGCM    58          I
          -CCG26
CCOMGCM   FWA      MACRO  NAM,EPN                                         CCOMGCM    59          I
          -CCG26
CCOMGCM            VFD    42/0L_NAM                                       CCOMGCM    60          I
          -CCG26
CCOMGCM            VFD    18/=XB=_EPN                                     CCOMGCM    61          I
          -CCG26
CCOMGCM            NOREF  B=_EPN                                          CCOMGCM    62          I
          -CCG26
CCOMGCM            ENDM                                                   CCOMGCM    63          I
          -CCG26
CCOMGCM   RPVFWA   SPACE  3,14                                            CCG26      31    A
CCOMGCM   **       RPVFWA - DEFINE ENTRY POINT FOR *RPV* NAME/ADDRESS TABLE.  CCG26  32    A
CCOMGCM   *                                                               CCG26      33    A
CCOMGCM   *        RPVFWA NAM,FWA                                         CCG26      34    A
CCOMGCM   *                                                               CCG26      35    A
CCOMGCM   *               NAM = ROUTINE NAME                              CCG26      36    A
CCOMGCM   *               FWA = ROUTINE FWA.  IF ABSENT, *B=XXXXX* IS USED WHERE  CCG26  37  A
CCOMGCM   *                     XXXXX ARE THE FIRST 5 CHARACTERS OF *NAM*.  CCG26  38    A
CCOMGCM                                                                   CCG26      39    A
CCOMGCM            PURGMAC  RPVFWA                                        CCG26      40    A
CCOMGCM                                                                   CCG26      41    A
CCOMGCM   RPVFWA   MACRO  NAM,FWA                                         CCG26      42    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCOMGCM

```
CCOMGCM            VFD    42/0L_NAM                                          CCG26        43     A
CCOMGCM            IFC    EQ,/FWA//,4                                        CCG26        44     A
CCOMGCM   '?RPV=A  MICRO  1,5,/NAM      /                                    CCG26        45     A
CCOMGCM            VFD    18/=XB="'?RPV=A"                                   CCG26        46     A
CCOMGCM            NOREF  B="'?RPV=A"                                        CCG26        47     A
CCOMGCM            SKIP   1                                                  CCG26        48     A
CCOMGCM            VFD    18/FWA                                             CCG26        49     A
CCOMGCM   RPVFWA   ENDM                                                      CCG26        50     A
CCOMGCM   LISTL    SPACE  4,8                                                CCOMGCM      64     A
CCOMGCM  **        LISTL - LIST ONE LINE.                                    CCOMGCM      65     A
CCOMGCM  *                                                                   CCOMGCM      66     A
CCOMGCM  *         WRITE ONE CODED LINE FROM *BUF* TO THE OUTPUT FILE.  IF PAGE  CCOMGCM  67     A
CCOMGCM  *         IS FULL, EJECT AND WRITE TITLE LINE(S) FIRST.             CCOMGCM      68     A
CCOMGCM  *                                                                   CCOMGCM      69     A
CCOMGCM  *                                                                   CCOMGCM      70     A
CCOMGCM  *         LISTL  BUF,WORDS,NBL                                      CCOMGCM      71     A
CCOMGCM  *                                                                   CCOMGCM      72     A
CCOMGCM  *         ENTRY  *BUF*   = FIRST WORD ADDRESS OF LINE BUFFER        CCOMGCM      73     A
CCOMGCM  *                *WORDS* = LINE LENGTH (WORDS)                      CCOMGCM      74     A
CCOMGCM  *                *NBL* = NUMBER OF BLANK LINES TO BE PRINTED BEFORE CCOMGCM      75     A
CCOMGCM  *                        DATA IN LINE BUFFER.  IF NBL > LINES REMAINING  CCOMGCM  76     A
CCOMGCM  *                        THEN A PAGE EJECT IS DONE.                 CCOMGCM      77     A
CCOMGCM  *                                                                   CCOMGCM      78     A
CCOMGCM  *         USES   X6, B6, B7                                         CCOMGCM      79     A
CCOMGCM  *                                                                   CCOMGCM      80     A
CCOMGCM  *         CALLS  FA=LOL ( CCOMLOL )                                 CCOMGCM      81     A
CCOMGCM                                                                      CCOMGCM      82     A
CCOMGCM                                                                      CCOMGCM      83     A
CCOMGCM            PURGMAC   LISTL                                           CCOMGCM      84     A
CCOMGCM                                                                      CCOMGCM      85     A
CCOMGCM   LISTL    MACRO  S,N,NBL                                            CCOMGCM      86     A
CCOMGCM            R=     B6,S                                               CCOMGCM      87     A
CCOMGCM  *                                                                   CCOMGCM      88     A
CCOMGCM            IFC    EQ, N  ,1                                          CCOMGCM      89     A
CCOMGCM            ERR    NO LENGTH "SEQUENCE"                               CCOMGCM      90     A
CCOMGCM  *                                                                   CCOMGCM      91     A
CCOMGCM            IFNE   CP#RM,0,1                                          CCOMGCM      92           I
          -CCG39
CCOMGCM            R=     B7,N                                               CCOMGCM      93     A
CCOMGCM  *                                                                   CCOMGCM      94     A
CCOMGCM            SX6    NBL 0                                              CCOMGCM      95     A
CCOMGCM  *                                                                   CCOMGCM      96     A
CCOMGCM            RJ     =XFA=LOL                                           CCOMGCM      97     A
CCOMGCM   LISTL    ENDM                                                      CCOMGCM      98     A
CCOMGCM   NUPAGE   SPACE  4,8                                                CCOMGCM      99     A
CCOMGCM  **        NUPAGE - EJECT AND TITLE NEW PAGE.                        CCOMGCM     100     A
CCOMGCM  *                                                                   CCOMGCM     101     A
CCOMGCM  *         EJECTS PAGE.  WRITES TITLE AND SUBTITLE LINES ON NEW PAGE. CCOMGCM     102     A
CCOMGCM  *         RESETS LINES-PER-PAGE COUNTER.                            CCOMGCM     103     A
CCOMGCM  *                                                                   CCOMGCM     104     A
CCOMGCM  *                                                                   CCOMGCM     105     A
CCOMGCM  *         NUPAGE    (NO CALLING PARAMETERS)                         CCOMGCM     106     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCOMGCM

      CCOMGCM  *                                                          CCOMGCM      107     A
      CCOMGCM  *         ENTRY  NO REQUIREMENTS.                          CCOMGCM      108     A
      CCOMGCM  *                                                          CCOMGCM      109     A
      CCOMGCM  *         EXIT   PAGE EJECTED, NEW PAGE TITLED.            CCOMGCM      110     A
      CCOMGCM  *                *N.LINES* RESET TO FULL PAGE LINE COUNT.  CCOMGCM      111     A
      CCOMGCM  *                                                          CCOMGCM      112     A
      CCOMGCM  *         USES   NONE                                      CCOMGCM      113     A
      CCOMGCM  *                                                          CCOMGCM      114     A
      CCOMGCM  *         CALLS  FA=NPG ( CCOMLOL )                        CCOMGCM      115     A
      CCOMGCM                                                             CCOMGCM      116     A
      CCOMGCM                                                             CCOMGCM      117     A
      CCOMGCM            PURGMAC   NUPAGE                                 CCOMGCM      118     A
      CCOMGCM                                                             CCOMGCM      119     A
      CCOMGCM   NUPAGE   MACRO                                            CCOMGCM      120     A
      CCOMGCM            RJ     =XFA=NPG                                  CCOMGCM      121     A
      CCOMGCM   NUPAGE   ENDM                                             CCOMGCM      122     A
      CCOMGCM                                                             CCOMGCM      123     A
      CCOMGCM                                                             CCOMGCM      124     A
      CCOMGCM            ENDX                                             CCOMGCM      125     A

           SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - CCOMGCM

                IDENTIFIER                TOTAL    ACTIVE

                CCOMGCM                     125        93
                CCG26                        44        44

                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCOMISC

      CCOMISC  *COMDECK  CCOMISC                                          CCOMISC        1     A
      CCOMISC   SETCORE  CTEXT  CCOMISC - INITIALIZE SMALL CORE ( SETCORE ).  CCOMISC      2     A
      CCOMISC                                                             CCOMISC        3     A
      CCOMISC            IF     -DEF,QUAL$,1                              CCOMISC        4     A
      CCOMISC            QUAL   COMCISC                                   CCOMISC        5     A
      CCOMISC  *         COMMENT COPYRIGHT CONTROL DATA CORP. 1976        CCOMISC        6          I
                -CCG1*92
      CCOMISC  *         COMMENT COPYRIGHT CONTROL DATA SYSTEMS INC. 1992 CCG1*92        9          I
                -CCG*94
      CCOMISC  *         COMMENT COPYRIGHT CONTROL DATA SYSTEMS, INC. 1994.  CCG*94      7     A
      CCOMISC   ISC=     SPACE  3,8                                       CCOMISC        7     A
      CCOMISC  ***       ISC - INITIALIZE A BLOCK OF SCM TO A GIVEN VALUE CCOMISC        8     A
      CCOMISC  *                                                          CCOMISC        9     A
      CCOMISC  *                                                          CCOMISC       10     A
      CCOMISC  *         ENTRY  (X1) = BLOCK LENGTH                       CCOMISC       11     A
      CCOMISC  *                (X6) = VALUE TO SET                       CCOMISC       12     A
      CCOMISC  *                (A6) = BLOCK FWA (WORD 1 ALREADY STORED)  CCOMISC       13     A
      CCOMISC  *                                                          CCOMISC       14     A
      CCOMISC  *         EXIT   (X6) = UNCHANGED                          CCOMISC       15     A
      CCOMISC  *                (X7) = (X6)                               CCOMISC       16     A
      CCOMISC  *                (A6) = BLOCK LWA                          CCOMISC       17     A
      CCOMISC  *                (B1) = 1                                  CCOMISC       18     A
      CCOMISC  *                                                          CCOMISC       19     A

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCOMISC

```
CCOMISC  *         USES   X - 1, 2, 7                                    CCOMISC      20     A
CCOMISC  *                A - 6, 7                                       CCOMISC      21     A
CCOMISC  *                B - 1                                          CCOMISC      22     A
CCOMISC  *                                                               CCOMISC      23     A
CCOMISC  *         CALLS  NONE                                           CCOMISC      24     A
CCOMISC                                                                  CCOMISC      25     A
CCOMISC                                                                  CCOMISC      26     A
CCOMISC   SETC2    SA7    A6+1                                           CCOMISC      27     A
CCOMISC            IX1    X1-X2                                          CCOMISC      28     A
CCOMISC            SA6    A7+B1                                          CCOMISC      29     A
CCOMISC            PL     X1,SETC2    IF ALL OF BLOCK NOT SET            CCOMISC      30     A
CCOMISC                                                                  CCOMISC      31     A
CCOMISC   ISC=     PS                 ENTRY/EXIT                         CCOMISC      32     A
CCOMISC                                                                  CCOMISC      33     A
CCOMISC            IFNE   TEST,0,2                                       CCOMISC      34     A
CCOMISC            SB1    X1+                                            CCOMISC      35     A
CCOMISC            LE     B1,*+1S17   ABORT JOB IF LENGTH ERROR          CCOMISC      36     A
CCOMISC                                                                  CCOMISC      37     A
CCOMISC            SX2    1                                              CCOMISC      38     A
CCOMISC            NO                                                    CCOMISC      39     A
CCOMISC            BX7    -X1*X2      EXTRACT COMPLEMENT OF LSB          CCOMISC      40     A
CCOMISC            SB1    X7          (B1) = 1 IF LENGTH EVEN, = 0 IF ODD CCOMISC     41     A
CCOMISC            SX1    X1-3        DECREMENT LENGTH                   CCOMISC      42     A
CCOMISC            SA6    A6+B1       SET 2ND WORD IF LENGTH WAS EVEN    CCOMISC      43     A
CCOMISC            LX2    1           (X2) = 2                           CCOMISC      44     A
CCOMISC            SB1    1                                              CCOMISC      45     A
CCOMISC            BX7    X6                                             CCOMISC      46     A
CCOMISC            PL     X1,SETC2    IF ALL OF BLOCK NOT SET            CCOMISC      47     A
CCOMISC            EQ     ISC=                                           CCOMISC      48     A
CCOMISC                                                                  CCOMISC      49     A
CCOMISC            IF     -DEF,QUAL$,2                                   CCOMISC      50     A
CCOMISC            QUAL   *                                              CCOMISC      51     A
CCOMISC   ISC=     EQU    /COMCISC/ISC=                                  CCOMISC      52     A
CCOMISC                                                                  CCOMISC      53     A
CCOMISC            ENDX                                                  CCOMISC      54     A
```

SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - CCOMISC

IDENTIFIER	TOTAL	ACTIVE
CCOMISC	54	53
CCG1*92	1	0
CCG*94	1	1

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCOMLOL

```
CCOMLOL  *COMDECK  CCOMLOL            LIST ONE LINE                      CCOMLOL       1     A
CCOMLOL            CTEXT  CCOMLOL - LIST ONE LINE TO THE OUTPUT FILE     CCOMLOL       2     A
CCOMLOL   LOL      SPACE  3                                              CCOMLOL       3     A
CCOMLOL            IF     -DEF,QUAL$,1                                   CCOMLOL       4     A
CCOMLOL            QUAL   CCOMLOL                                        CCOMLOL       5     A
CCOMLOL            BASE   D                                              CCOMLOL       6     A
```

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCOMLOL

```
CCOMLOL  *         COMMENT COPYRIGHT CONTROL DATA CORP. 1976                CCOMLOL      7        I
         -CCG*94
CCOMLOL  *         COMMENT COPYRIGHT CONTROL DATA SYSTEMS, INC. 1994.       CCG*94       8     A
CCOMLOL   LOL      SPACE  3                                                 CCOMLOL      8     A
CCOMLOL  ***       LOL - LIST ONE LINE                                      CCOMLOL      9     A
CCOMLOL  *         D. C. DILLON.   72/5/21.                                 CCOMLOL     10     A
CCOMLOL   LOL      SPACE  3                                                 CCOMLOL     11     A
CCOMLOL  ***              FA=LOL TRANSFER 1 CODED LINE IN -C- FORMAT FROM A CCOMLOL     12     A
CCOMLOL  *         WORKING BUFFER TO A CIO BUFFER WITH LINE AND PAGE CONTROL. CCOMLOL     13     A
CCOMLOL  *         A TITLE LINE WILL BE PLACED AT THE TOP OF EACH PAGE AND THE CCOMLOL     14     A
CCOMLOL  *         PAGE NUMBER UPDATED.                                     CCOMLOL     15     A
CCOMLOL  *                FA=NPG EJECTS, TITLES AND SUBTITLES A NEW PAGE.   CCOMLOL     16     A
CCOMLOL  *                O.STITL IS THE FWA OF A 12 WORD BUFFER USED TO HOLD CCOMLOL     17     A
CCOMLOL  *                THE CURRENT SUBTITLE.                             CCOMLOL     18     A
CCOMLOL  *                L.STITL HOLDS THE LENGTH OF THE CURRENT SUBTITLE. CCOMLOL     19     A
CCOMLOL  *                N.LINES IS MINUS THE NUMBER OF LINES REMAINING TO BE CCOMLOL     20     A
CCOMLOL  *                PRINTED ON THE CURRENT PAGE.                      CCOMLOL     21     A
CCOMLOL  *                                                                  CCOMLOL     22     A
CCOMLOL  *         THE CALLER IS EXPECTED TO SATISFY THE FOLLOWING -        CCOMLOL     23     A
CCOMLOL  *         HO$BL - NON ZERO IF -BL- SELECTED ON CONTROL STMT.       CCOMLOL     24     A
CCOMLOL  *         CP.PAGE - LAST PAGE NUMBER ( BINARY INTEGER ).           CCOMLOL     25     A
CCOMLOL  *         CP.PS - NEGATIVE OF NUMBER OF LINES/PAGE.                CCOMLOL     26     A
CCOMLOL  *         F.OUT - FWA OF OUTPUT FILE FET OR PSEUDO FET.            CCOMLOL     27     A
CCOMLOL  *         F$TITL - FWA OF TITLE LINE                               CCOMLOL     28     A
CCOMLOL  *         Z$TITL - LENGTH OF TITLE LINE ( AN EQU )                 CCOMLOL     29     A
CCOMLOL  *         TL.PAGE - PLACE IN TITLE LINE WHERE PAGE NUMBER IS PLACED IN CCOMLOL     30     A
CCOMLOL  *                6L NNN FORMAT. IT IS THE LAST WORD OF THE LINE.   CCOMLOL     31     A
CCOMLOL  *         CDD= - ROUTINE IN COMCCDD.                               CCOMLOL     32     A
CCOMLOL                                                                     CCOMLOL     33     A
CCOMLOL   N$STITL  DATA   1           LENGTH OF SUBTITLE LINE               CCOMLOL     34     A
CCOMLOL   F$STITL  DATA   8L          SUBTITLE LINE                         CCOMLOL     35     A
CCOMLOL            BSS    11                                                 CCOMLOL     36     A
CCOMLOL   N$LINES  DATA   0           N. LINES REMAINING ON THE PAGE        CCOMLOL     37     A
CCOMLOL                                                                     CCOMLOL     38     A
CCOMLOL  **        FA=LOL - LIST ONE LINE ON OUTPUT FILE                    CCOMLOL     39     A
CCOMLOL  *                                                                  CCOMLOL     40     A
CCOMLOL  *         ENTRY  (B6) = FWA OF LINE BUFFER                         CCOMLOL     41     A
CCOMLOL  *                (B7) = LINE LENGTH IN WORDS.                      CCOMLOL     42     A
CCOMLOL  *                (X6) = NUMBER OF BLANK LINES TO BE PRINTED PRIOR TO CCOMLOL     43     A
CCOMLOL  *                       PRINTING THE CURRENT LINE.                 CCOMLOL     44     A
CCOMLOL  *                (B1) = 1                                          CCOMLOL     45     A
CCOMLOL  *                                                                  CCOMLOL     46     A
CCOMLOL  *         PRESERVES - A0,X0, A5,X5                                 CCOMLOL     47     A
CCOMLOL                                                                     CCOMLOL     48     A
CCOMLOL   LOL2     WRITEC =XF.OUT,B6,B7                                     CCOMLOL     49     A
CCOMLOL                                                                     CCOMLOL     50     A
CCOMLOL   FA=LOL   PS                 ENTRY/EXIT                            CCOMLOL     51     A
CCOMLOL                                                                     CCOMLOL     52     A
CCOMLOL            ZR     X6,LOL3     IF NBL = 0                            CCOMLOL     53     A
CCOMLOL            IX7    X6+X1                                             CCOMLOL     54     A
CCOMLOL            SA7    A1                                                CCOMLOL     55     A
CCOMLOL            PL     X7,LOL3     IF N.LINES + NBL OVERFLOWS THE PAGE   CCOMLOL     56     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCOMLOL

```
CCOMLOL                                                                    CCOMLOL      57    A
CCOMLOL            SA6    LOLB                                             CCOMLOL      58    A
CCOMLOL            SX7    B6                                               CCOMLOL      59    A
CCOMLOL            IFNE   CP#RM,0,1                                        CCOMLOL      60    A
CCOMLOL            PX7    B7,X7                                            CCOMLOL      61    A
CCOMLOL            SA7    LOLA                                             CCOMLOL      62    A
CCOMLOL                                                                    CCOMLOL      63    A
CCOMLOL   LOL1     WRITEC F.OUT,(=8L  ),1                                  CCOMLOL      64    A
CCOMLOL            SA1    LOLB                                             CCOMLOL      65    A
CCOMLOL            SX6    X1-1                                             CCOMLOL      66    A
CCOMLOL            SA6    A1                                               CCOMLOL      67    A
CCOMLOL            NZ     X6,LOL1                                          CCOMLOL      68    A
CCOMLOL                                                                    CCOMLOL      69    A
CCOMLOL            SA4    LOLA                                             CCOMLOL      70    A
CCOMLOL            SB6    X4                                               CCOMLOL      71    A
CCOMLOL            IFNE   CP#RM,0,1                                        CCOMLOL      72    A
CCOMLOL            UX7    B7,X4                                            CCOMLOL      73    A
CCOMLOL                                                                    CCOMLOL      74    A
CCOMLOL   LOL3     SA3    N$LINES                                          CCOMLOL      75    A
CCOMLOL            SX6    X1+B1       DECREMENT LINE COUNT                 CCOMLOL      76    A
CCOMLOL            SA6    A1                                               CCOMLOL      77    A
CCOMLOL            MI     X6,LOL2     IF PAGE NOT FULL                     CCOMLOL      78    A
CCOMLOL            SX6    B6                                               CCOMLOL      79    A
CCOMLOL            IFNE   CP#RM,0,1                                        CCOMLOL      80    A
CCOMLOL            PX6    B7,X6                                            CCOMLOL      81    A
CCOMLOL            SA6    LOLA        SAVE LINE FWA AND LENGTH             CCOMLOL      82    A
CCOMLOL            RJ     FA=NPG      EJECT AND TITLE PAGE                 CCOMLOL      83    A
CCOMLOL            SA2    LOLA                                             CCOMLOL      84    A
CCOMLOL            SA1    N$LINES                                          CCOMLOL      85    A
CCOMLOL            SB6    X2                                               CCOMLOL      86    A
CCOMLOL            SX6    X1+B1       DECREMENT LINE COUNT                 CCOMLOL      87    A
CCOMLOL            IFNE   CP#RM,0,1                                        CCOMLOL      88    A
CCOMLOL            UX2    B7,X2                                            CCOMLOL      89    A
CCOMLOL            SA6    A1                                               CCOMLOL      90    A
CCOMLOL            EQ     LOL2                                             CCOMLOL      91    A
CCOMLOL                                                                    CCOMLOL      92    A
CCOMLOL    LOLA    BSS    1                                                CCOMLOL      93    A
CCOMLOL    LOLB    BSS    1                                                CCOMLOL      94    A
CCOMLOL    FA=NPG  SPACE  3,14                                             CCOMLOL      95    A
CCOMLOL   **       FA=NPG - EJECT AND TITLE NEW PAGE.                      CCOMLOL      96    A
CCOMLOL   *                                                                CCOMLOL      97    A
CCOMLOL   *               IF *BL* OPTION ON, UNCONDITIONALLY EJECTS, TITLES AND  CCOMLOL  98    A
CCOMLOL   *        SUBTITLES NEW PAGE.                                     CCOMLOL      99    A
CCOMLOL   *               IF *BL* OPTION OFF, AND 8 OR MORE LINES REMAIN ON  CCOMLOL  100   A
CCOMLOL   *        CURRENT PAGE, SPACES 4 AND LISTS SUBTITLE ONLY.  IF LESS THAN  CCOMLOL  101   A
CCOMLOL   *        8 LINES REMAIN, EJECTS, TITLES AND SUBTITLES NEW PAGE.  CCOMLOL  102   A
CCOMLOL   *        BEHAVES LIKE COMPASS *SPACE 4,4* DIRECTIVE.             CCOMLOL      103   A
CCOMLOL   *                                                                CCOMLOL      104   A
CCOMLOL   *                                                                CCOMLOL      105   A
CCOMLOL   *        ENTRY  (B1) = 1                                         CCOMLOL      106   A
CCOMLOL   *                                                                CCOMLOL      107   A
CCOMLOL   *        EXIT   PAGE SPACED OR EJECTED, AND TITLED.              CCOMLOL      108   A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCOMLOL

```
CCOMLOL  *                (N.LINES) ADJUSTED (IF SPACE 4) OR RESET (IF EJECT).          CCOMLOL    109    A
CCOMLOL  *                (CP.PAGE) INCREMENTED IF PAGE EJECTED.                        CCOMLOL    110    A
CCOMLOL  *                (B1) = 1                                                      CCOMLOL    111    A
CCOMLOL  *                                                                              CCOMLOL    112    A
CCOMLOL  *          PRESERVES A0,X0, A5,X5                                              CCOMLOL    113    A
CCOMLOL                                                                                 CCOMLOL    114    A
CCOMLOL   FA=NPG    PS                  ENTRY/EXIT                                      CCOMLOL    115    A
CCOMLOL             SA1    F$STITL                                                      CCOMLOL    116    A
CCOMLOL             SX2    1R0                                                          CCOMLOL    117    A
CCOMLOL             MX6    6                                                            CCOMLOL    118    A
CCOMLOL             LX2    -6                                                           CCOMLOL    119    A
CCOMLOL             BX1    -X6*X1                                                       CCOMLOL    120    A
CCOMLOL             IX6    X2+X1                                                        CCOMLOL    121    A
CCOMLOL             SA6    F$STITL     FWA OF SUBTITLE LINE                             CCOMLOL    122    A
CCOMLOL             SA1    =XHO$BL                                                      CCOMLOL    123    A
CCOMLOL             NZ     X1,NPG2     IF BURSTABLE LISTING                             CCOMLOL    124    A
CCOMLOL             SA1    N$LINES     CHECK SPACE LEFT ON CURRENT PAGE                 CCOMLOL    125    A
CCOMLOL             SX6    X1+8-1                                                       CCOMLOL    126    A
CCOMLOL             PL     X6,NPG2     IF LESS THAN 8 LINES LEFT                        CCOMLOL    127    A
CCOMLOL                                                                                 CCOMLOL    128    A
CCOMLOL  *          SPACE 4 AND SUBTITLE CURRENT PAGE.                                  CCOMLOL    129    A
CCOMLOL                                                                                 CCOMLOL    130    A
CCOMLOL             SX6    X1+4+1                                                       CCOMLOL    131    A
CCOMLOL             SA6    A1          UPDATE *LINES REMAINING* COUNT                   CCOMLOL    132    A
CCOMLOL             WRITEC F.OUT,(=1C-),1    TRIPLE SPACE                               CCOMLOL    133    A
CCOMLOL             EQ     NPG3                                                         CCOMLOL    134    A
CCOMLOL                                                                                 CCOMLOL    135    A
CCOMLOL  *          EJECT, TITLE AND SUBTITLE NEW PAGE.                                 CCOMLOL    136    A
CCOMLOL                                                                                 CCOMLOL    137    A
CCOMLOL   NPG2      SA1    =XCP.PAGE   LAST PAGE NUMBER (BINARY INTEGER)                CCOMLOL    138    A
CCOMLOL             MX6    -1                                                           CCOMLOL    139    A
CCOMLOL             IX7    X1-X6       PAGE NUMBER + 1                                  CCOMLOL    140    A
CCOMLOL             SX1    X7          REMOVE PROPAGATION BIT, IF PRESENT               CCOMLOL    141    A
CCOMLOL             SA7    A1                                                           CCOMLOL    142    A
CCOMLOL             RJ     =XCDD=      RETURNS (X6) = PAGE NUMBER IN DISPLAY CODE       CCOMLOL    143    A
CCOMLOL             SA1    CP.PS                                                        CCOMLOL    144    A
CCOMLOL             MX7    24                                                           CCOMLOL    145    A
CCOMLOL             BX6    -X7*X6                                                       CCOMLOL    146    A
CCOMLOL             LX6    24                                                           CCOMLOL    147    A
CCOMLOL             SA6    TL.PAGE     SET NEW PAGE NUMBER IN TITLE LINE                CCOMLOL    148    A
CCOMLOL             BX6    X1                                                           CCOMLOL    149    A
CCOMLOL             SA6    N$LINES                                                      CCOMLOL    150    A
CCOMLOL                                                                                 CCOMLOL    151    A
CCOMLOL  *          LIST TITLE AND SUBTITLE LINES                                       CCOMLOL    152    A
CCOMLOL                                                                                 CCOMLOL    153    A
CCOMLOL             WRITEC F.OUT,F$TITL,Z$TITL                                          CCOMLOL    154    A
CCOMLOL   NPG3      SA1    L$STITL                                                      CCOMLOL    155    A
CCOMLOL             WRITEC =XF.OUT,F$STITL,X1                                           CCOMLOL    156    A
CCOMLOL                                                                                 CCOMLOL    157    A
CCOMLOL             EQ     FA=NPG                                                       CCOMLOL    158    A
CCOMLOL                                                                                 CCOMLOL    159    A
CCOMLOL             BASE   *                                                            CCOMLOL    160    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCOMLOL

CCOMLOL            IF     -DEF,QUAL$                                          CCOMLOL      161     A
CCOMLOL            QUAL   *                                                   CCOMLOL      162     A
CCOMLOL            ECHO   1,Z=(F$STITL,N$STITL,N$LINES,FA=LOL,FA=NPG)         CCOMLOL      163     A
CCOMLOL    Z       EQU    /CCOMLOL/Z                                          CCOMLOL      164     A
CCOMLOL            ENDIF                                                      CCOMLOL      165     A
CCOMLOL                                                                       CCOMLOL      166     A
CCOMLOL            IF     -ABS,FA=LOL,2                                       CCOMLOL      167     A
CCOMLOL            ECHO   1,Z=(F$STITL,N$STITL,N$LINES,FA=LOL,FA=NPG)         CCOMLOL      168     A
CCOMLOL            ENTRY  Z                                                   CCOMLOL      169     A
CCOMLOL                                                                       CCOMLOL      170     A
CCOMLOL            ENDX                                                       CCOMLOL      171     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - CCOMLOL

          IDENTIFIER                TOTAL    ACTIVE

          CCOMLOL                     171       170
          CCG*94                        1         1

          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGHCRMD

CGHCRMD  *COMDECK  CGHCRMD                                                    CGHCRMD        1     A
CGHCRMD   CGHCRMD  CTEXT  CGHCRMD - HOST COMPILER REQUIRED MACRO DEFINITIONS  CGHCRMD        2     A
CGHCRMD  **        CGHCRMD - HOST COMPILER REQUIRED MACRO DEFINITIONS FOR THE CGHCRMD        3     A
CGHCRMD  *         HOST COMPILER"S ASSEMBLY TEXT.                             CGHCRMD        4     A
CGHCRMD  *                                                                    CGHCRMD        5     A
CGHCRMD  *         IF THE HOST MAKES THE *C* OPTION ( COMPASS ASSEMBLY )      CGHCRMD        6     A
CGHCRMD  *         AVAILABLE, THEN IT WILL HAVE TO SUPPLY A "LPN"MAC SYSTEMS  CGHCRMD        7     A
CGHCRMD  *         TEXT FOR COMPASS TO USE WHEN THE *COMPS* FILE IS ASSEMBLED. CGHCRMD        8     A
CGHCRMD  *         THE BELOW MACROS SHOULD BE INCLUDED IN THAT TEXT.          CGHCRMD        9     A
CGHCRMD   USKBLK   SPACE  2,12                                                CGHCRMD       10     A
CGHCRMD  **        USEBLK - DEFINE LIBRARY NAMES,ORDERING OF FIXED LOCAL BLOCK CGHCRMD     11     A
CGHCRMD  *         AND OTHER MISCELLANEOUS VALUES FOR USE AT ASSEMBLY TIME    CGHCRMD       12     A
CGHCRMD                                                                       CGHCRMD       13     A
CGHCRMD   USEBLK   MACRO                                                      CGHCRMD       14     A
CGHCRMD            SST                                                        CGHCRMD       15     A
CGHCRMD  *                         DEFINE ORDER OF FIXED LOCAL BLOCKS         CGHCRMD       16     A
CGHCRMD            IFC    NE,/"HC.FLB"//,4                                    CGHCRMD       17     A
CGHCRMD            ECHO   1,Z=("HC.FLB")                                      CGHCRMD       18     A
CGHCRMD            USE    Z                                                   CGHCRMD       19     A
CGHCRMD  *                         INITIALIZE K. FOR ORG MACRO                CGHCRMD       20     A
CGHCRMD            USE    "HC.UDVB"                                           CGHCRMD       21     A
CGHCRMD   K.       SET    *                                                   CGHCRMD       22     A
CGHCRMD  *                         START OFF IN THE FIRST BLOCK               CGHCRMD       23     A
CGHCRMD   FB       MICRO  1,,,"HC.FLB",                                       CGHCRMD       24     A
CGHCRMD            USE    "FB"                                                CGHCRMD       25     A
CGHCRMD  *                         DEFINE EXECUTION TIME CPU                  CGHCRMD       26     A
CGHCRMD   .CPU     =      "MDL"                                               CGHCRMD       27     A
CGHCRMD            ENDM                                                       CGHCRMD       28     A
CGHCRMD   LIB=     SPACE  3,14                                                CGHCRMD       29     A
CGHCRMD  **        LIB= - DEFINE LIBRARY NAMES                                CGHCRMD       30     A
CGHCRMD                                                                       CGHCRMD       31     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGHCRMD

```
CGHCRMD   LIB=     MACRO  A                                                    CGHCRMD      32    A
CGHCRMD            LDSET  LIB=A                                                CGHCRMD      33    A
CGHCRMD            ENDM                                                        CGHCRMD      34    A
CGHCRMD   RJQ      SPACE  2,10                                                 CGHCRMD      35    A
CGHCRMD  *         RJQ - REDEFINE THE RJ INSTRUCTION SO IT IS A DEFAULT EXTERNAL CGHCRMD   36    A
CGHCRMD  *         REFERENCE.                                                  CGHCRMD      37    A
CGHCRMD                                                                        CGHCRMD      38    A
CGHCRMD   ZJQ      CPSYN  RJQ                                                  CGHCRMD      39    A
CGHCRMD            PURGDEF RJQ                                                 CGHCRMD      40    A
CGHCRMD   RJQ      OPDEF  Q                                                    CGHCRMD      41    A
CGHCRMD    ZJ =X;A                                                             CGHCRMD      42    A
CGHCRMD            ENDM                                                        CGHCRMD      43    A
CGHCRMD   RJT      SPACE  2,12                                                 CGHCRMD      44    A
CGHCRMD  **        RJT - RETURN JUMP WITH TRACEBACK                            CGHCRMD      45    A
CGHCRMD                                                                        CGHCRMD      46    A
CGHCRMD   RJT      MACRO  SYM,LINE                                             CGHCRMD      47    A
CGHCRMD  +         RJ     SYM                                                  CGHCRMD      48    A
CGHCRMD            IF     DEF,SYM_(,1                                          CGHCRMD      49    A
CGHCRMD            SUB    SYM                                                  CGHCRMD      50    A
CGHCRMD            IFLT   LINE,7777B,2                                         CGHCRMD      51    A
CGHCRMD  -         VFD    12/LINE,18/"HC.RJTBN"                                CGHCRMD      52    A
CGHCRMD            ELSE   3                                                    CGHCRMD      53    A
CGHCRMD  -         VFD    12/7777B,18/"HC.RJTBN"                               CGHCRMD      54    A
CGHCRMD            SB0    0                                                    CGHCRMD      55    A
CGHCRMD            SB0    LINE                                                 CGHCRMD      56    A
CGHCRMD            ENDM                                                        CGHCRMD      57    A
CGHCRMD   RJXJ     SPACE  2,12                                                 CGHCRMD      58    A
CGHCRMD  **        RJXJ - DEFINE CONDITIONAL RETURN JUMP MACROS                CGHCRMD      59    A
CGHCRMD  *         RJXX   XI,SYM+CA                                            CGHCRMD      60    A
CGHCRMD                                                                        CGHCRMD      61    A
CGHCRMD            ECHO   4,XX=("XJUMPS"),YY=("^XJUMPS")                       CGHCRMD      62    A
CGHCRMD   RJ_XX    MACRO  XR,SYM                                               CGHCRMD      63    A
CGHCRMD  +         YY     XR,*+1                                               CGHCRMD      64    A
CGHCRMD            RJ     SYM                                                  CGHCRMD      65    A
CGHCRMD            ENDM                                                        CGHCRMD      66    A
CGHCRMD   ORG      SPACE  2,14                                                 CGHCRMD      67    A
CGHCRMD  **        ORG - REDEFINE *ORG* PSEUDO FOR USE OF *CCG*                CGHCRMD      68    A
CGHCRMD  *                                                                     CGHCRMD      69    A
CGHCRMD  *         ORG    IH,CA                                                CGHCRMD      70    A
CGHCRMD                                                                        CGHCRMD      71    A
CGHCRMD   ORG.     OPSYN  ORG                                                  CGHCRMD      72    A
CGHCRMD            PURGMAC  ORG                                                CGHCRMD      73    A
CGHCRMD                                                                        CGHCRMD      74    A
CGHCRMD   ORG      MACRO  IH,CA                                                CGHCRMD      75    A
CGHCRMD   O        IFC    NE,//IH/                                             CGHCRMD      76    A
CGHCRMD            USE    "HC.UDVB"                                            CGHCRMD      77    A
CGHCRMD   I        IF     -DEF,IH                                              CGHCRMD      78    A
CGHCRMD   IH       BSS    0                                                    CGHCRMD      79    A
CGHCRMD   I        ELSE                                                        CGHCRMD      80    A
CGHCRMD   K.       MAX    K.,*        SAVE CURRENT VALUE OF ORG COUNTER        CGHCRMD      81    A
CGHCRMD            ORG.   IH+CA       SET NEW ORIGIN                           CGHCRMD      82    A
CGHCRMD   S.       SET    *           FWA FOR DATA PLACEMENT ( USED BY *REPI* ) CGHCRMD     83    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGHCRMD

```
CGHCRMD   I        ENDIF                                                    CGHCRMD      84    A
CGHCRMD   O        ELSE                                                     CGHCRMD      85    A
CGHCRMD            ORG    K.          RESTORE PREVIOUS VALUE OF ORG COUNTER CGHCRMD      86    A
CGHCRMD            ENDIF                                                    CGHCRMD      87    A
CGHCRMD            ENDM                                                     CGHCRMD      88    A
CGHCRMD   REPI     SPACE  2,14                                             CGHCRMD      89    A
CGHCRMD  **        REPI - REDEFINE *REPI* PSEUDO FOR USE BY *CCG*          CGHCRMD      90    A
CGHCRMD  *                                                                 CGHCRMD      91    A
CGHCRMD  *         REPI   DATA-LEN,REP-COUNT,INCREMENT,DESTIN-ADDR         CGHCRMD      92    A
CGHCRMD                                                                    CGHCRMD      93    A
CGHCRMD   REPI.    OPSYN  REPI                                             CGHCRMD      94    A
CGHCRMD            PURGMAC  REPI                                           CGHCRMD      95    A
CGHCRMD                                                                    CGHCRMD      96    A
CGHCRMD   REPI     MACRO  DL,RC,INC,DA                                     CGHCRMD      97    A
CGHCRMD            LOCAL  II                                               CGHCRMD      98    A
CGHCRMD   II       SET    INC 0                                            CGHCRMD      99    A
CGHCRMD            REPI.  S/S.,B/DL,C/RC,I/II,D/S.+DA                      CGHCRMD     100    A
CGHCRMD            ENDM                                                    CGHCRMD     101    A
CGHCRMD                                                                    CGHCRMD     102    A
CGHCRMD   #FPAS    IFNE   HC$FPAS,0         F.P. ADDRESS SUBSTITUTION MACROS CGHCRMD     103    A
CGHCRMD   FORPAR   SPACE  2,10                                             CGHCRMD     104    A
CGHCRMD  **        FORPAR - DEFINE ORDER OF *FP* USE BLOCK FOR *SUB* WORDS. CGHCRMD     105    A
CGHCRMD                                                                    CGHCRMD     106    A
CGHCRMD   FORPAR   MACRO  FP                                               CGHCRMD     107    A
CGHCRMD            USE    FP                                               CGHCRMD     108    A
CGHCRMD   FP.      BSS    0                                                CGHCRMD     109    A
CGHCRMD   FP_(     SET    0                                                CGHCRMD     110    A
CGHCRMD            ENDM                                                    CGHCRMD     111    A
CGHCRMD   SUB0     SPACE  2,10                                             CGHCRMD     112    A
CGHCRMD  **        SUB0 - SUB MACRO FOR A LEVEL 0 PARAMETER ( SCM OR LCM ) CGHCRMD     113    A
CGHCRMD  *                                                                 CGHCRMD     114    A
CGHCRMD  *         SUB0   RIJ,FP                                           CGHCRMD     115    A
CGHCRMD  *                RIJ = 3/R,3/I,3/J WHERE                          CGHCRMD     116    A
CGHCRMD  *                           R = 0/1 FOR READ/WRITE                CGHCRMD     117    A
CGHCRMD  *                           I,J = RESULT AND ADDRESS REGISTER NUMBERS CGHCRMD 118    A
CGHCRMD                                                                    CGHCRMD     119    A
CGHCRMD   SUB0     MACRO  RIJ,FP                                           CGHCRMD     120    A
CGHCRMD   .P       SET    59-$                                             CGHCRMD     121    A
CGHCRMD   .O       SET    *-$/59                                           CGHCRMD     122    A
CGHCRMD            USE    FP                                               CGHCRMD     123    A
CGHCRMD   IJ       MICRO  2,,/RIJ/                                         CGHCRMD     124    A
CGHCRMD  *                           CHANGE TO 6000B-.P OR SUCH ( WHATEVER WORKS CGHCRMD 125    A
CGHCRMD            VFD    12/1S10+.P,15/01400B+RIJ,15/53"IJ"0B,18/.O       CGHCRMD     126    A
CGHCRMD            IFEQ   FP_(,.O,1                                        CGHCRMD     127    A
CGHCRMD            VFD    12/2036B,48/ST.   DELAY WORD                     CGHCRMD     128    A
CGHCRMD   FP_(     SET    .O                                               CGHCRMD     129    A
CGHCRMD            USE    *                                                CGHCRMD     130    A
CGHCRMD            ENDM                                                    CGHCRMD     131    A
CGHCRMD   SUB      SPACE  2,10                                             CGHCRMD     132    A
CGHCRMD  *         SUB - FORM AN ADDRESS SUBSTITUTION WORD FOR A FORMAL PARAM. CGHCRMD 133    A
CGHCRMD                                                                    CGHCRMD     134    A
CGHCRMD   SUB      MACRO  CA,FP                                            CGHCRMD     135    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGHCRMD

CGHCRMD   .P       SET    59-$                                                  CGHCRMD      136     A
CGHCRMD   .O       SET    *-$/59                                                CGHCRMD      137     A
CGHCRMD            USE    FP                                                    CGHCRMD      138     A
CGHCRMD            VFD    12/2S9+.P,12/,18/CA,18/.O                             CGHCRMD      139     A
CGHCRMD            IFEQ   FP_(,.O,1                                             CGHCRMD      140     A
CGHCRMD            VFD    12/2036B,48/ST.   DELAY WORD                          CGHCRMD      141     A
CGHCRMD   FP_(     SET    .O                                                    CGHCRMD      142     A
CGHCRMD            USE    *                                                     CGHCRMD      143     A
CGHCRMD            ENDM                                                         CGHCRMD      144     A
CGHCRMD                                                                         CGHCRMD      145     A
CGHCRMD   #FPAS    ENDIF                                                        CGHCRMD      146     A
CGHCRMD                                                                         CGHCRMD      147     A
CGHCRMD            ENDX                                                         CGHCRMD      148     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - CGHCRMD

                IDENTIFIER                TOTAL    ACTIVE

                CGHCRMD                     148       148

                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMADEF

COMADEF  *COMDECK  COMADEF - STRUCTURED FIELD DECLARATION MACROS.               COMADEF        1     A
COMADEF   COMADEF  CTEXT  COMADEF - STRUCTURED FIELD DECLARATION MACROS.        COMADEF        2     A
COMADEF  ***       COMADEF - STRUCTURED FIELD DECLARATION MACROS.               COMADEF        3     A
COMADEF  *                                                                      COMADEF        4     A
COMADEF  *         J. A. WHITE III AND S. I. JASIK - DATES UNKNOWN              COMADEF        5     A
COMADEF  *         P. H. MCQUESTEN    76-10-06.                                 COMADEF        6     A
COMADEF  *                                                                      COMADEF        7     A
COMADEF  *         MACROS TO FACILITATE FIELD DESCRIPTION, WHERE FIELDS ARE     COMADEF        8     A
COMADEF  *         SUB-WORD ENTITIES.  THE 'DESCRIBE' MACRO IS USED TO PROVIDE  COMADEF        9     A
COMADEF  *         A PREFIX FOR NAMES SUPPLIED IN SUBSEQUENT 'DEFINE' REFERENCES. COMADEF       10     A
COMADEF  *         FOR EACH NAME SPECIFIED, THE FOLLOWING SYMBOLS WILL BE       COMADEF       11     A
COMADEF  *         DEFINED                                                      COMADEF       12     A
COMADEF  *                                                                      COMADEF       13     A
COMADEF  *               PFX_NAME_P -- THE BIT POSITION OF THE RIGHT MOST BIT   COMADEF       14     A
COMADEF  *                             WHICH COMPRISES THE NAMED FIELD (WITHIN  COMADEF       15     A
COMADEF  *                             A WORD BITS ARE NUMBERED ACCORDING TO    COMADEF       16     A
COMADEF  *                             THE POWER OF TWO WHICH THEY REPRESENT)   COMADEF       17     A
COMADEF  *               PFX_NAME_L -- THE LENGTH IN BITS OF THE FIELD          COMADEF       18     A
COMADEF  *                                                                      COMADEF       19     A
COMADEF  *         FIELDS ARE NOT PERMITTED TO SPAN WORD BOUNDARIES (A FATAL-TO- COMADEF       20     A
COMADEF  *         ASSEMBLY ERROR WILL RESULT) OR TO GO BEYOND THE TOTAL NUMBER COMADEF       21     A
COMADEF  *         OF BITS WHICH THE ENTIRE SET OF FIELDS IS SUPPOSED TO OCCUPY. COMADEF       22     A
COMADEF  *                                                                      COMADEF       23     A
COMADEF  *         EACH 'DEFINE' REFERENCES A FIELD BEGINNING TO THE IMMEDIATE  COMADEF       24     A
COMADEF  *         RIGHT OF THE PRECEEDING FIELD.                               COMADEF       25     A
COMADEF   DESCRIBE SPACE  3                                                     COMADEF       26     A
COMADEF  ***       DESCRIBE - DECLARE BEGINNING OF STRUCTURE.                   COMADEF       27     A
COMADEF  *                                                                      COMADEF       28     A
COMADEF  *         DESCRIBE PFX,BITSLONG,TOPBIT,WORDNR                          COMADEF       29     A
COMADEF  *                                                                      COMADEF       30     A

                0        1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMADEF

```
COMADEF  *         ENTRY  PFX = THE PREFIX FOR THE STRUCTURE, AS MENTIONED ABOVE.        COMADEF    31    A
COMADEF  *                BITSLONG = THE TOTAL LENGTH IN BITS OF THE STRUCTURE,          COMADEF    32    A
COMADEF  *                     IF NOT PRESENT A VALUE OF 60 IS ASSUMED.                  COMADEF    33    A
COMADEF  *                TOPBIT = THE BEGINNING (LEFTMOST) BIT OF THE STRUCTURE,        COMADEF    34    A
COMADEF  *                     IF ABSENT "BITSLONG-1" IS THE DEFAULT VALUE.              COMADEF    35    A
COMADEF  *                WORDNR = WORD NUMBER OF THE STRUCTURE BEING DESCRIBE-D.        COMADEF    36    A
COMADEF  *                     ZERO IF OMITTED.  SHOULD BE OMITTED FOR                   COMADEF    37    A
COMADEF  *                     SINGLE-WORD STRUCTURES.                                   COMADEF    38    A
COMADEF                                                                                  COMADEF    39    A
COMADEF                                                                                  COMADEF    40    A
COMADEF   DESCRIBE MACRO  PREFIX,BITSLONG,TOPBIT,WORDNR                                  COMADEF    41    A
COMADEF            NOREF  /"QUAL"/.C,/"QUAL"/.D,/"QUAL"/.W                               COMADEF    42    A
COMADEF            NOREF  /"QUAL"/.B,/"QUAL"/.L,/"QUAL"/.1                               COMADEF    43    A
COMADEF            NOREF  /"QUAL"/.2,/"QUAL"/.3,/"QUAL"/.4                               COMADEF    44    A
COMADEF            NOREF  /"QUAL"/.5                                                     COMADEF    45    A
COMADEF   .P       MICRO  1,,/PREFIX/                                                    COMADEF    46    A
COMADEF   .D       SET    BITSLONG   60                                                  COMADEF    47    A
COMADEF   .C       SET    TOPBIT     .D-1                                                COMADEF    48    A
COMADEF   .W       SET    WORDNR     0                                                   COMADEF    49    A
COMADEF  *                           INITIALIZE COUNTERS.                                COMADEF    50    A
COMADEF   .L       SET    .D                                                             COMADEF    51    A
COMADEF   .B       SET    .C                                                             COMADEF    52    A
COMADEF            IFC    NE, WORDNR  ,1                                                 COMADEF    53    A
COMADEF   ".P"W    EQU    .W                                                             COMADEF    54    A
COMADEF   DESCRIBE ENDM                                                                  COMADEF    55    A
COMADEF   DEFINE   SPACE  4,20                                                           COMADEF    56    A
COMADEF  ***       DEFINE - DECLARE A FIELD WITHIN THE STRUCTURE.                        COMADEF    57    A
COMADEF  *                                                                               COMADEF    58    A
COMADEF  *  NAME   DEFINE LENGTH,R                                                       COMADEF    59    A
COMADEF  *                                                                               COMADEF    60    A
COMADEF  *         ENTRY  NAME = THE NAME OF THE FIELD MENTIONED ABOVE.  IF NOT          COMADEF    61    A
COMADEF  *                     PRESENT THEN THE REFERENCE WILL ACT AS FILLER.            COMADEF    62    A
COMADEF  *                LENGTH = THE BIT LENGTH OF THIS FIELD (OR FILLER).             COMADEF    63    A
COMADEF  *                     IF OMITTED, LENGTH = ONE IS ASSUMED.                      COMADEF    64    A
COMADEF  *                R = IF PRESENT, WILL CAUSE THE NAMED FIELD TO BE               COMADEF    65    A
COMADEF  *                     DEFINED RELATIVE TO BIT POSITION 'R' AS THE               COMADEF    66    A
COMADEF  *                     TOPMOST (LEFTMOST) BIT.  THE 'REDEF' MACRO WILL           COMADEF    67    A
COMADEF                                                                                  COMADEF    68    A
COMADEF                                                                                  COMADEF    69    A
COMADEF            MACRO  DEFINE,N,LONG,RESET                                            COMADEF    70    A
COMADEF   .1       SET    LONG 1                                                         COMADEF    71    A
COMADEF            IFC    NE,//RESET/,2                                                  COMADEF    72    A
COMADEF   .L       SET    1000    ** KLUDGE **      .L-.B+RESET                          COMADEF    73    A
COMADEF   .B       SET    RESET                                                          COMADEF    74    A
COMADEF  *                                                                               COMADEF    75    A
COMADEF            IFC    NE,  N ,1                                                      COMADEF    76    A
COMADEF   N        DEQU   *,.1,.B-.1+1                                                   COMADEF    77    A
COMADEF  *                           ADVANCE COUNTERS, CHECK OVERFLOW.                   COMADEF    78    A
COMADEF   .L       SET    .L-.1                                                          COMADEF    79    A
COMADEF   .B       SET    .B-.1                                                          COMADEF    80    A
COMADEF            ERRMI  .L          TOTAL LENGTH OF ".P" LONGER THAN SPECIFIED.        COMADEF    81    A
COMADEF            IFLT   .B,0,3                                                         COMADEF    82    A
```

```
         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMADEF

```
COMADEF            IFNE   .B,-1,1                                                  COMADEF      83    A
COMADEF             ERR    FIELD SPANS A WORD BOUNDARY    ("SEQUENCE")             COMADEF      84    A
COMADEF   .B       SET    59                                                       COMADEF      85    A
COMADEF   DEFINE   ENDM                                                            COMADEF      86    A
COMADEF   DEQU     SPACE  4,20                                                     COMADEF      87    A
COMADEF  ***       DEQU - DECLARE EQUIVALENCED FIELD.                              COMADEF      88    A
COMADEF  *                                                                         COMADEF      89    A
COMADEF  *  NAM    DEQU   FLD,LEN                                                  COMADEF      90    A
COMADEF  *                                                                         COMADEF      91    A
COMADEF  *         ENTRY  NAM = A NEW FIELD NAME TO BE DECLARED.                   COMADEF      92    A
COMADEF  *               FLD = A PREVIOUSLY DEFINED FIELD IN THIS STRUCTURE.       COMADEF      93    A
COMADEF  *               LEN = (OPTIONAL) EXPLICIT BIT LENGTH OF NEW FIELD.        COMADEF      94    A
COMADEF  *                     IF 'LEN' IS OMITTED, LENGTH OF THE NEW FIELD        COMADEF      95    A
COMADEF  *                     WILL BE SAME AS LENGTH OF OLD FIELD.                COMADEF      96    A
COMADEF                                                                            COMADEF      97    A
COMADEF                                                                            COMADEF      98    A
COMADEF            MACRO  DEQU,N,B,LEN,V                                           COMADEF      99    A
COMADEF   .2       SET    LEN  ".P"B_L                                            COMADEF     100    A
COMADEF   .3       SET    V    ".P"B_P                                            COMADEF     101    A
COMADEF   ".P"N_P  EQU    .3                                                      COMADEF     102    A
COMADEF   ".P"N_L  EQU    .2                                                      COMADEF     103    A
COMADEF  *                          LIST FIELD MASK PRETTILY.                      COMADEF     104    A
COMADEF            LOC    .W*100B+.3                                              COMADEF     105    A
COMADEF            POS    1+.C                                                    COMADEF     106    A
COMADEF   .4       SET    *P-.2-.3                                                COMADEF     107    A
COMADEF   .5       SET    .D-.2-.4                                                COMADEF     108    A
COMADEF            VFD    .4/,.2/-0,.5/                                           COMADEF     109    A
COMADEF            ORG    *O-*P/60D                                               COMADEF     110    A
COMADEF  *                          DEFINE MASK VALUE FOR SINGLE BIT FIELDS.       COMADEF     111    A
COMADEF            IFEQ   .2,1,4                                                  COMADEF     112    A
COMADEF            NOREF  /"QUAL"/".P"N_L                                         COMADEF     113    A
COMADEF            IFLE   .3,17,2                                                 COMADEF     114    A
COMADEF   .A       DECMIC .3                                                      COMADEF     115    A
COMADEF   ".P"N_M  EQU    1S".A"                                                  COMADEF     116    A
COMADEF            ENDM                                                           COMADEF     117    A
COMADEF   REDEF    SPACE  4,10                                                    COMADEF     118    A
COMADEF  ***       REDEF - DECLARE OVERLAPPING SUB-STRUCTURES.                    COMADEF     119    A
COMADEF  *                                                                        COMADEF     120    A
COMADEF  *         REDEF  NAME                                                    COMADEF     121    A
COMADEF  *                                                                        COMADEF     122    A
COMADEF  *         ENTRY  NAME = THE NAME OF A PREVIOUSLY DECLARED FIELD.         COMADEF     123    A
COMADEF  *                                                                        COMADEF     124    A
COMADEF  *         CAUSES SUBSEQUENT FIELD TO BE DEFINED BEGINNING AT THE TOP     COMADEF     125    A
COMADEF  *         (LEFTMOST) BIT OF FIELD 'NAME'.  IF 'NAME' IS OMITTED,         COMADEF     126    A
COMADEF  *         REDEFINITION BEGINS AT THE TOP OF THE ENTIRE STRUCTURE.        COMADEF     127    A
COMADEF                                                                           COMADEF     128    A
COMADEF                                                                           COMADEF     129    A
COMADEF   REDEF    MACRO  N                                                       COMADEF     130    A
COMADEF            IFC    EQ, N  ,3                                               COMADEF     131    A
COMADEF   .1       SET    .C                                                      COMADEF     132    A
COMADEF   .L       SET    .D                                                      COMADEF     133    A
COMADEF            SKIP   2                                                       COMADEF     134    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMADEF

```
COMADEF   .1       SET    ".P"N_P+".P"N_L-1                                  COMADEF      135     A
COMADEF   .L       SET    1000  ** KLUDGE **  .L+.1-.B+1                     COMADEF      136     A
COMADEF  *                                                                   COMADEF      137     A
COMADEF   .B       SET    .1                                                 COMADEF      138     A
COMADEF            ENDM                                                      COMADEF      139     A
COMADEF   EXAMPLE  SPACE  4,30                                               COMADEF      140     A
COMADEF  ***       CONSIDER THE FOLLOWING EXAMPLE --                         COMADEF      141     A
COMADEF  *                                                                   COMADEF      142     A
COMADEF  *         TABLE X      .------.-------.------.                      COMADEF      143     A
COMADEF  *                      . TYPE   VALUE . SUBV .  (SAY FOR TYPE @ 100) COMADEF      144     A
COMADEF  *                      .------.-------.------.                      COMADEF      145     A
COMADEF  *                       '      '       '      '                      COMADEF      146     A
COMADEF  *                      59     47      17     0                      COMADEF      147     A
COMADEF  *                                                                   COMADEF      148     A
COMADEF  *                OR    .------.--------------.                      COMADEF      149     A
COMADEF  *                      . TYPE . MESSAGE      .  (FOR TYPE > 100)    COMADEF      150     A
COMADEF  *                      .------.--------------.                      COMADEF      151     A
COMADEF  *                       '      '              '                      COMADEF      152     A
COMADEF  *                      59     47             0                      COMADEF      153     A
COMADEF  *                                                                   COMADEF      154     A
COMADEF  *                DESCRIBE X.,60    TABLE X, THE PREFIX TO BE USED IS THE COMADEF 155     A
COMADEF  *                                  TWO CHARACTERS 'X.'              COMADEF      156     A
COMADEF  *         TYPE   DEFINE  12                                         COMADEF      157     A
COMADEF  *         MSG    DEFINE  48                                         COMADEF      158     A
COMADEF  *                REDEF   MSG                                        COMADEF      159     A
COMADEF  *         VALUE  DEFINE  30                                         COMADEF      160     A
COMADEF  *         SUBV   DEFINE  18                                         COMADEF      161     A
COMADEF  *                                                                   COMADEF      162     A
COMADEF  *       THE SYMBOLS DEFINED WOULD BE                                COMADEF      163     A
COMADEF  *                                                                   COMADEF      164     A
COMADEF  *                X.TYPEP   EQU  48                                  COMADEF      165     A
COMADEF  *                X.TYPEL   EQU  12                                  COMADEF      166     A
COMADEF  *                X.MSGP    EQU   0                                  COMADEF      167     A
COMADEF  *                X.MSGL    EQU  48                                  COMADEF      168     A
COMADEF  *                X.VALUEP  EQU  18                                  COMADEF      169     A
COMADEF  *                X.VALUEL  EQU  30                                  COMADEF      170     A
COMADEF  *                X.SUBVP   EQU   0                                  COMADEF      171     A
COMADEF  *                X.SUBVL   EQU  18                                  COMADEF      172     A
COMADEF   BMFW     SPACE  4,20                                               COMADEF      173     A
COMADEF  ***       BFMW - CREATE A BIT FIELD MASK WORD.                      COMADEF      174     A
COMADEF  *                                                                   COMADEF      175     A
COMADEF  * LAB     BFMW   PREFIX,(BITLIST),SHFT                              COMADEF      176     A
COMADEF  *                                                                   COMADEF      177     A
COMADEF  *         ENTRY  PREFIX = BIT FIELD PREFIX CHARACTERS WITH NO PERIOD. COMADEF     178     A
COMADEF  *                BITLIST = LIST OF BIT FIELDS TO APPEAR IN MASK.    COMADEF      179     A
COMADEF  *                SHFT = AMOUNT BY WHICH WORD IN REGISTERS HAS ALREADY COMADEF    180     A
COMADEF  *                       BEEN SHIFTED.                               COMADEF      181     A
COMADEF                                                                      COMADEF      182     A
COMADEF                                                                      COMADEF      183     A
COMADEF            MACRO  BFMW,LAB,PR,BIT,SHFT                               COMADEF      184     A
COMADEF            IFC    NE,/"QUAL"//,1                                     COMADEF      185     A
COMADEF            NOREF  /"QUAL"/.P,/"QUAL"/.Z                              COMADEF      186     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMADEF

COMADEF  *                                                                      COMADEF      187     A
COMADEF   LAB      BSS    0                                                    COMADEF      188     A
COMADEF   .Z       SET    0                                                    COMADEF      189     A
COMADEF                                                                         COMADEF      190     A
COMADEF            IRP    BIT                                                  COMADEF      191     A
COMADEF   .P       SET    PR.BIT_P+PR.BIT_L+SHFT                               COMADEF      192     A
COMADEF            IFGT   .P,60,1                                              COMADEF      193     A
COMADEF   .P       SET    .P-60                                                COMADEF      194     A
COMADEF            IFNE   .P,PR.BIT_L,3                                        COMADEF      195     A
COMADEF            POS    .P                                                   COMADEF      196     A
COMADEF            VFD    PR.BIT_L/-0                                          COMADEF      197     A
COMADEF            ELSE   1                                                    COMADEF      198     A
COMADEF   .Z       SET    .P                                                   COMADEF      199     A
COMADEF            IRP                                                         COMADEF      200     A
COMADEF  *                                                                      COMADEF      201     A
COMADEF            IFNE   .Z,0,3                                               COMADEF      202     A
COMADEF            POS    .Z                                                   COMADEF      203     A
COMADEF            VFD    .Z/-0                                                COMADEF      204     A
COMADEF            SKIP   1                                                    COMADEF      205     A
COMADEF            VFD    *P/0                                                 COMADEF      206     A
COMADEF  *                                                                      COMADEF      207     A
COMADEF            ENDM                                                        COMADEF      208     A
COMADEF                                                                         COMADEF      209     A
COMADEF   COMADEF  ENDX                                                         COMADEF      210     A

      SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - COMADEF

                IDENTIFIER                TOTAL    ACTIVE

                COMADEF                     210       210

                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMAIDP

COMAIDP  *COMDECK COMAIDP                                                 000150   COMAIDP        1     A
COMAIDP   COMAIDP  CTEXT  COMAIDP - INTERACTIVE AND BATCH DEBUG PACKAGE MACROS.  000160   COMAIDP        2     A
COMAIDP   COMAIDP  SPACE  4,8                                             000170   COMAIDP        3     A
COMAIDP            IF     -DEF,QUAL$,1                                    000180   COMAIDP        4     A
COMAIDP            QUAL   COMAIDP                                         000190   COMAIDP        5     A
COMAIDP            BASE   D                                               000200   COMAIDP        6     A
COMAIDP   COMAIDP  SPACE  4,8                                             000210   COMAIDP        7     A
COMAIDP  **        COMAIDP - INTERACTIVE AND BATCH DEBUG PACKAGE MACROS.  000220   COMAIDP        8     A
COMAIDP  *                                                                000230   COMAIDP        9     A
COMAIDP  *                                                                000240   COMAIDP       10     A
COMAIDP  *                CONTAINS THE MACROS USED TO ACCESS THE ROUTINES IN  000250   COMAIDP       11     A
COMAIDP  *         *COMCIDP*--                                            000260   COMAIDP       12     A
COMAIDP  *                                                                000270   COMAIDP       13     A
COMAIDP  * NAME    BREAK  LL,UL,INC   TO PLACE A BREAKPOINT               000280   COMAIDP       14     A
COMAIDP  * NAME    REG                TO PROVIDE A SNAPSHOT OF ALL REGISTERS  000290   COMAIDP       15     A
COMAIDP  * NAME    REG    (X1,X2,A4)  TO PROVIDE A SNAPSHOT OF SELECTED REGISTERS000300   COMAIDP       16     A
COMAIDP  * NAME    SNAP   FWA,LWA,LEN TO PROVIDE A SNAPSHOT OF CENTRAL MEMORY  000310   COMAIDP       17     A
COMAIDP   BREAK    SPACE  4,8                                             000320   COMAIDP       18     A
COMAIDP  **        BREAK - PLACE A BREAKPOINT.                            000330   COMAIDP       19     A

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMAIDP

```
COMAIDP  *                                                          000340  COMAIDP   20   A
COMAIDP  *                                                          000350  COMAIDP   21   A
COMAIDP  *               THIS MACRO DOES A RETURN JUMP TO THE INTERACTIVE  000360  COMAIDP   22   A
COMAIDP  *         DEBUG PACKAGE *IDP*.                             000370  COMAIDP   23   A
COMAIDP  *                                                          000380  COMAIDP   24   A
COMAIDP  * NAME    BREAK  LL,UL,INC                                 000390  COMAIDP   25   A
COMAIDP  *                                                          000400  COMAIDP   26   A
COMAIDP  *         ENTRY  NAME = DISPLAY CODE (DPC) MESSAGE TO BE PRINTED WITH  000410  COMAIDP   27   A
COMAIDP  *                       EACH SNAP. (NR CHARS.LE.8)         000420  COMAIDP   28   A
COMAIDP  *               LL   = LOWER LIMIT (DEFAULT=1)             000430  COMAIDP   29   A
COMAIDP  *               UL   = UPPER LIMIT (DEFAULT=100)           000440  COMAIDP   30   A
COMAIDP  *               NUSE =  USE BLOCK FLAG. IF *NUSE* IS NULL, THE PARA-  000450  COMAIDP   31   A
COMAIDP  *                         METER LIST EXISTS IN *USE DEBUG*, ELSE IS  000460  COMAIDP   32   A
COMAIDP  *                         ASSEMBLED AFTER *RJ*             000470  COMAIDP   33   A
COMAIDP  *               UF   = A STRING OF USER SNAP FLAGS PROCESSED BY        COMAIDP   34   A
COMAIDP  *                      INSTALLER SUPPLIED USF= MACRO IF DESIRED.       COMAIDP   35   A
COMAIDP  *                      USF= SHOULD GENERATE 1 WORD.                    COMAIDP   36   A
COMAIDP  *                                                          000480  COMAIDP   37   A
COMAIDP  *         EXIT   NONE                                      000490  COMAIDP   38   A
COMAIDP  *                                                          000500  COMAIDP   39   A
COMAIDP  *         USES   NONE                                      000510  COMAIDP   40   A
COMAIDP  *                                                          000520  COMAIDP   41   A
COMAIDP  *         CALLS  IDP=                                      000530  COMAIDP   42   A
COMAIDP                                                             000540  COMAIDP   43   A
COMAIDP                                                             000550  COMAIDP   44   A
COMAIDP            PURGMAC BREAK                                    000560  COMAIDP   45   A
COMAIDP                                                             000570  COMAIDP   46   A
COMAIDP            MACRO  BREAK,NAME,LL,UL,INC,NUSE,UF                     COMAIDP   47   A
COMAIDP            LOCAL  '?ID#001,'?ID#004                                COMAIDP   48   A
COMAIDP            LOCAL  RTN                                       000590  COMAIDP   49   A
COMAIDP            LOCAL  APL                                       000600  COMAIDP   50   A
COMAIDP  *                                                          000610  COMAIDP   51   A
COMAIDP   +        RJ     =XIDP=                                    000620  COMAIDP   52   A
COMAIDP   -        VFD    30/APL                                    000630  COMAIDP   53   A
COMAIDP  *                                                          000640  COMAIDP   54   A
COMAIDP   #NUSE    IFC    EQ,/NUSE//                                000650  COMAIDP   55   A
COMAIDP            USE    DEBUG                                     000660  COMAIDP   56   A
COMAIDP  *                                                          000670  COMAIDP   57   A
COMAIDP   #NUSE    ELSE                                             000680  COMAIDP   58   A
COMAIDP            EQ     RTN                                       000690  COMAIDP   59   A
COMAIDP   #NUSE    ENDIF                                            000700  COMAIDP   60   A
COMAIDP   APL      BSS    0                                         000710  COMAIDP   61   A
COMAIDP            FRK=   (NAME),LL,UL,INC                          000720  COMAIDP   62   A
COMAIDP  *                                                                  COMAIDP   63   A
COMAIDP   '?ID#001 BSS 0                                                    COMAIDP   64   A
COMAIDP            USF=   (UF)                                              COMAIDP   65   A
COMAIDP  *                                                                  COMAIDP   66   A
COMAIDP   '?ID#002 IFNE *P,60D                                              COMAIDP   67   A
COMAIDP            IFNE *P,0,2                                              COMAIDP   68   A
COMAIDP            POS 0                                                    COMAIDP   69   A
COMAIDP   '?ID#003 SKIP                                                     COMAIDP   70   A
COMAIDP  *                                                                  COMAIDP   71   A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMAIDP

```
COMAIDP   '?ID#002 ENDIF                                                          COMAIDP      72     A
COMAIDP   '?ID#004 BSS 0                                                          COMAIDP      73     A
COMAIDP  *                                                                        COMAIDP      74     A
COMAIDP            IFNE   '?ID#004-'?ID#001,1,2                                   COMAIDP      75     A
COMAIDP   '?ID#003 ENDIF                                                          COMAIDP      76     A
COMAIDP   P        ERR    *USF=* DID NOT GENERATE EXACTLY 60D BITS. "SEQUENCE"   COMAIDP      77     A
COMAIDP  *                                                                        COMAIDP      78     A
COMAIDP            CON    0                                                      COMAIDP      79     A
COMAIDP  *                                                             000730     COMAIDP      80     A
COMAIDP   #NUSE    IFC    EQ,/NUSE//                                     000740     COMAIDP      81     A
COMAIDP            USE    *                                              000750     COMAIDP      82     A
COMAIDP  *                                                             000760     COMAIDP      83     A
COMAIDP   #NUSE    ELSE                                                   000770     COMAIDP      84     A
COMAIDP   RTN      BSS    0                                              000780     COMAIDP      85     A
COMAIDP   #NUSE    ENDIF                                                  000790     COMAIDP      86     A
COMAIDP  *                                                             000800     COMAIDP      87     A
COMAIDP   BREAK    ENDM                                                   000810     COMAIDP      88     A
COMAIDP   FRK=     SPACE  4,8                                            000820     COMAIDP      89     A
COMAIDP  **        FRK= - GENERATE FREQUENCY PARAMETER LIST.              000830     COMAIDP      90     A
COMAIDP  *                                                             000840     COMAIDP      91     A
COMAIDP  *                                                             000850     COMAIDP      92     A
COMAIDP  *         FRK=   (NAME),LL,UL,INC                                000860     COMAIDP      93     A
COMAIDP  *                                                             000870     COMAIDP      94     A
COMAIDP  *         ENTRY  NAME =  DISPLAY CODE (DPC) MESSAGE TO BE PRINTED WITH  000880     COMAIDP      95     A
COMAIDP  *                          EACH SNAP. (NR CHARS.LE.8)         000890     COMAIDP      96     A
COMAIDP  *                LL   =  LOWER LIMIT (DEFAULT=1)              000900     COMAIDP      97     A
COMAIDP  *                UL   =  UPPER LIMIT (DEFAULT=100)            000910     COMAIDP      98     A
COMAIDP  *                INC  =  INCREMENT (DEFAULT=1)                000920     COMAIDP      99     A
COMAIDP  *                          LL, UL, AND INC ARE THE SNAP FREQUENCY PAR-  000930     COMAIDP     100     A
COMAIDP  *                          AMETERS AND ARE RELATED AS FOLLOWS--  THE 1ST000940     COMAIDP     101     A
COMAIDP  *                          SNAP PRINTED WILL OCCUR THE LLTH TIME THIS   000950     COMAIDP     102     A
COMAIDP  *                          SNAP IS EXECUTED. THE 2ND SNAP PRINTED WILL  000960     COMAIDP     103     A
COMAIDP  *                          OCCUR THE (LL+INC)TH TIME THIS SNAP IS EXEC- 000970     COMAIDP     104     A
COMAIDP  *                          UTED, AND SO ON. THE LAST SNAP PRINTED WILL  000980     COMAIDP     105     A
COMAIDP  *                          OCCUR THE ULTH TIME THIS SNAP IS EXECUTED.   000990     COMAIDP     106     A
COMAIDP  *                                                             001000     COMAIDP     107     A
COMAIDP  *         EXIT   NONE                                            001010     COMAIDP     108     A
COMAIDP  *                                                             001020     COMAIDP     109     A
COMAIDP  *         USES   NONE                                            001030     COMAIDP     110     A
COMAIDP  *                                                             001040     COMAIDP     111     A
COMAIDP  *         CALLS  NONE                                            001050     COMAIDP     112     A
COMAIDP                                                                   001060     COMAIDP     113     A
COMAIDP                                                                   001070     COMAIDP     114     A
COMAIDP            PURGMAC FRK=                                           001080     COMAIDP     115     A
COMAIDP                                                                   001090     COMAIDP     116     A
COMAIDP   FRK=     MACRO  NAME,LL,UL,INC                                  001100     COMAIDP     117     A
COMAIDP            CON    LL  1                                           001110     COMAIDP     118     A
COMAIDP            CON    UL  100                                         001120     COMAIDP     119     A
COMAIDP            CON    INC 1                                           001130     COMAIDP     120     A
COMAIDP   .1       IFC    EQ,/NAME//                                     001140     COMAIDP     121     A
COMAIDP            CON    0                                              001150     COMAIDP     122     A
COMAIDP   .1       ELSE                                                   001160     COMAIDP     123     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMAIDP

```
COMAIDP            CON    10H_NAME                                          001170   COMAIDP      124     A
COMAIDP   .1       ENDIF                                                    001180   COMAIDP      125     A
COMAIDP            CON    0                                                 001190   COMAIDP      126     A
COMAIDP   *                          USED BY *FRK=/TSNAP* TO KEEP SNAP COUNT 001200   COMAIDP      127     A
COMAIDP    FRK=    ENDM                                                     001210   COMAIDP      128     A
COMAIDP    REG     SPACE  4,8                                               001220   COMAIDP      129     A
COMAIDP   **       REG - REGISTER SNAPSHOT.                                 001230   COMAIDP      130     A
COMAIDP   *                                                                 001240   COMAIDP      131     A
COMAIDP   *                                                                 001250   COMAIDP      132     A
COMAIDP   * NAME   REG    RG,LL,UL,INC,NUSE,UF                              001260   COMAIDP      133     A
COMAIDP   *                                                                 001270   COMAIDP      134     A
COMAIDP   *        REG                                                      001280   COMAIDP      135     A
COMAIDP   *        REG    (X1,B3,A4)                                        001290   COMAIDP      136     A
COMAIDP   *        REG    NOREG                                             001300   COMAIDP      137     A
COMAIDP   *                                                                 001310   COMAIDP      138     A
COMAIDP   *        ENTRY  NAME =  DISPLAY CODE (DPC) MESSAGE TO BE PRINTED WITH 001320   COMAIDP      139     A
COMAIDP   *                         EACH SNAP. (NR CHARS.LE.8)              001330   COMAIDP      140     A
COMAIDP   *                     =  NULL IF NO SNAP HEADER                   001340   COMAIDP      141     A
COMAIDP   *                RG   =  NULL IF ALL REGISTERS ARE TO BE DUMPED   001350   COMAIDP      142     A
COMAIDP   *                     =  A LIST OF THE REGISTERS (ENCLOSED IN PARENS) 001360   COMAIDP      143     A
COMAIDP   *                         DESIRED TO BE DUMPED                    001370   COMAIDP      144     A
COMAIDP   *                     =  NON-REGISTER/NON-NULL IF NO REGISTERS TO BE 001380   COMAIDP      145     A
COMAIDP   *                         DUMPED                                  001390   COMAIDP      146     A
COMAIDP   *                LL   =  LOWER LIMIT (DEFAULT=1)                  001400   COMAIDP      147     A
COMAIDP   *                UL   =  UPPER LIMIT (DEFAULT=100)                001410   COMAIDP      148     A
COMAIDP   *                NUSE =  USE BLOCK FLAG. IF *NUSE* IS NULL, THE PARA- 001420   COMAIDP      149     A
COMAIDP   *                         METER LIST EXISTS IN *USE DEBUG*, ELSE IS 001430   COMAIDP      150     A
COMAIDP   *                         ASSEMBLED AFTER *RJ*                    001440   COMAIDP      151     A
COMAIDP   *                UF   =  A STRING OF USER SNAP FLAGS PROCESSED BY 001450   COMAIDP      152     A
COMAIDP   *                        INSTALLER SUPPLIED USF= MACRO IF DESIRED. 001460   COMAIDP      153     A
COMAIDP   *                        USF= SHOULD GENERATE 1 WORD.             001470   COMAIDP      154     A
COMAIDP   *                                                                 001480   COMAIDP      155     A
COMAIDP   *        EXIT   NONE                                              001490   COMAIDP      156     A
COMAIDP   *                                                                 001500   COMAIDP      157     A
COMAIDP   *        USES   NONE                                              001510   COMAIDP      158     A
COMAIDP   *                                                                 001520   COMAIDP      159     A
COMAIDP   *        CALLS  FRK=,REG=,RGR=,USF=(IF DEF)                       001530   COMAIDP      160     A
COMAIDP                                                                     001540   COMAIDP      161     A
COMAIDP                                                                     001550   COMAIDP      162     A
COMAIDP            PURGMAC REG                                              001560   COMAIDP      163     A
COMAIDP                                                                     001570   COMAIDP      164     A
COMAIDP            MACRO  REG,NAME,RG,LL,UL,INC,NUSE,UF                     001580   COMAIDP      165     A
COMAIDP            LOCAL  '?ID#001,'?ID#004                                 001590   COMAIDP      166     A
COMAIDP            LOCAL  RTN                                               001600   COMAIDP      167     A
COMAIDP            LOCAL  APL                                               001610   COMAIDP      168     A
COMAIDP   *                                                                 001620   COMAIDP      169     A
COMAIDP    +       RJ     =XREG=                                            001630   COMAIDP      170     A
COMAIDP    -       VFD    30/APL                                            001640   COMAIDP      171     A
COMAIDP   *                                                                 001650   COMAIDP      172     A
COMAIDP    #NUSE   IFC    EQ,/NUSE//                                        001660   COMAIDP      173     A
COMAIDP            USE    DEBUG                                             001670   COMAIDP      174     A
COMAIDP   *                                                                 001680   COMAIDP      175     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMAIDP

```
COMAIDP   #NUSE    ELSE                                                    001690   COMAIDP      176     A
COMAIDP            EQ     RTN                                              001700   COMAIDP      177     A
COMAIDP   #NUSE    ENDIF                                                   001710   COMAIDP      178     A
COMAIDP   APL      BSS    0                                                001720   COMAIDP      179     A
COMAIDP            FRK=   (NAME),LL,UL,INC                                 001730   COMAIDP      180     A
COMAIDP   '?ID#001 BSS 0                                                   001740   COMAIDP      181     A
COMAIDP            USF=   (UF)                                             001750   COMAIDP      182     A
COMAIDP   *                                                                001760   COMAIDP      183     A
COMAIDP   '?ID#002 IFNE *P,60D                                             001770   COMAIDP      184     A
COMAIDP            IFNE *P,0,2                                             001780   COMAIDP      185     A
COMAIDP            POS 0                                                   001790   COMAIDP      186     A
COMAIDP   '?ID#003 SKIP                                                    001800   COMAIDP      187     A
COMAIDP   *                                                                001810   COMAIDP      188     A
COMAIDP   '?ID#002 ENDIF                                                   001820   COMAIDP      189     A
COMAIDP   '?ID#004 BSS 0                                                   001830   COMAIDP      190     A
COMAIDP   *                                                                001840   COMAIDP      191     A
COMAIDP            IFNE   '?ID#004-'?ID#001,1,2                            001850   COMAIDP      192     A
COMAIDP   '?ID#003 ENDIF                                                   001860   COMAIDP      193     A
COMAIDP   P        ERR    *USF=* DID NOT GENERATE EXACTLY 60D BITS. "SEQUENCE"  001870   COMAIDP      194     A
COMAIDP   *                                                                001880   COMAIDP      195     A
COMAIDP            RGR=   (RG)                                             001890   COMAIDP      196     A
COMAIDP   *                                                                001900   COMAIDP      197     A
COMAIDP   #NUSE    IFC    EQ,/NUSE//                                      001910   COMAIDP      198     A
COMAIDP            USE    *                                               001920   COMAIDP      199     A
COMAIDP   *                                                                001930   COMAIDP      200     A
COMAIDP   #NUSE    ELSE                                                    001940   COMAIDP      201     A
COMAIDP   RTN      BSS    0                                               001950   COMAIDP      202     A
COMAIDP   #NUSE    ENDIF                                                   001960   COMAIDP      203     A
COMAIDP   *                                                                001970   COMAIDP      204     A
COMAIDP   REG      ENDM                                                    001980   COMAIDP      205     A
COMAIDP                                                                    001990   COMAIDP      206     A
COMAIDP            PURGMAC REGS                                            002000   COMAIDP      207     A
COMAIDP   REGS     OPSYN  REG                                              002010   COMAIDP      208     A
COMAIDP   RGR=     SPACE  4,8                                              002020   COMAIDP      209     A
COMAIDP   **       RGR= - GENERATE REGISTER PARAMETER LIST.                002030   COMAIDP      210     A
COMAIDP   *                                                                002040   COMAIDP      211     A
COMAIDP   *                                                                002050   COMAIDP      212     A
COMAIDP   *                SEE *DSR/COMCIDP*.                              002060   COMAIDP      213     A
COMAIDP   *                                                                002070   COMAIDP      214     A
COMAIDP   *        RGR=    RG                                              002080   COMAIDP      215     A
COMAIDP   *                                                                002090   COMAIDP      216     A
COMAIDP   *        RGR=                                                    002100   COMAIDP      217     A
COMAIDP   *        RGR=    (X1,X2,A3,B4)                                   002110   COMAIDP      218     A
COMAIDP   *        RGR=    NOREG                                           002120   COMAIDP      219     A
COMAIDP   *                                                                002130   COMAIDP      220     A
COMAIDP   *        ENTRY   RG    =  NULL IF ALL REGISTERS ARE TO BE DUMPED 002140   COMAIDP      221     A
COMAIDP   *                      =  A LIST OF THE REGISTERS (ENCLOSED IN PARENS)  002150   COMAIDP      222     A
COMAIDP   *                         DESIRED TO BE DUMPED                   002160   COMAIDP      223     A
COMAIDP   *                      =  NON-REGISTER/NON-NULL IF NO REGISTERS TO BE  002170   COMAIDP      224     A
COMAIDP   *                         DUMPED                                 002180   COMAIDP      225     A
COMAIDP   *                                                                002190   COMAIDP      226     A
COMAIDP   *        EXIT    NONE                                            002200   COMAIDP      227     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMAIDP

```
COMAIDP  *                                                          002210  COMAIDP   228   A
COMAIDP  *         USES   NONE                                      002220  COMAIDP   229   A
COMAIDP  *                                                          002230  COMAIDP   230   A
COMAIDP  *         CALLS  NONE                                      002240  COMAIDP   231   A
COMAIDP                                                             002250  COMAIDP   232   A
COMAIDP                                                             002260  COMAIDP   233   A
COMAIDP            PURGMAC RGR=                                     002270  COMAIDP   234   A
COMAIDP                                                             002280  COMAIDP   235   A
COMAIDP   RGR=     MACRO  RG                                        002290  COMAIDP   236   A
COMAIDP  *                                                          002300  COMAIDP   237   A
COMAIDP   .1       IFC    EQ,/RG//                                  002310  COMAIDP   238   A
COMAIDP            DATA   -1                                        002320  COMAIDP   239   A
COMAIDP   .1       ELSE                                             002330  COMAIDP   240   A
COMAIDP  *                                                          002340  COMAIDP   241   A
COMAIDP   .2       IRP    RG                                        002350  COMAIDP   242   A
COMAIDP   .3       IF     -REG,RG                                   002360  COMAIDP   243   A
COMAIDP            DATA   0                                         002370  COMAIDP   244   A
COMAIDP   .3       ELSE                                             002380  COMAIDP   245   A
COMAIDP   .T       MICRO  1,1,/RG/                                  002390  COMAIDP   246   A
COMAIDP   .R       MICRO  2,1,/RG/                                  002400  COMAIDP   247   A
COMAIDP            VFD    60/=XIDP=SV_".T"+".R"                     002410  COMAIDP   248   A
COMAIDP   .3       ENDIF                                            002420  COMAIDP   249   A
COMAIDP   .2       IRP                                              002430  COMAIDP   250   A
COMAIDP  *                                                          002440  COMAIDP   251   A
COMAIDP            DATA   0                                         002450  COMAIDP   252   A
COMAIDP   .1       ENDIF                                            002460  COMAIDP   253   A
COMAIDP  *                                                          002470  COMAIDP   254   A
COMAIDP   RGR=     ENDM                                             002480  COMAIDP   255   A
COMAIDP   SNAP     SPACE  4,8                                       002490  COMAIDP   256   A
COMAIDP  **        SNAP - SNAPSHOT OF REGISTERS AND CORE.           002500  COMAIDP   257   A
COMAIDP  *                                                          002510  COMAIDP   258   A
COMAIDP  *                                                          002520  COMAIDP   259   A
COMAIDP  * NAME    SNAP   FWA,LWA,LNG,RG,LL,UL,INC,NUSE,UF           002530  COMAIDP   260   A
COMAIDP  *         SNAP   *FWA,,**LNG,(X1,X2,B5,A1)                 002540  COMAIDP   261   A
COMAIDP  *                                                          002550  COMAIDP   262   A
COMAIDP  *         ENTRY  NAME =  DISPLAY CODE (DPC) MESSAGE TO BE PRINTED WITH  002560  COMAIDP   263   A
COMAIDP  *                          EACH SNAP. (NR CHARS.LE.8)      002570  COMAIDP   264   A
COMAIDP  *                     =  NULL IF NO SNAP HEADER            002580  COMAIDP   265   A
COMAIDP  *                FWA  =  FWA OF CORE TO BE DUMPED. THIS PARAMETER MUST  002590  COMAIDP   266   A
COMAIDP  *                          APPEAR IF A CORE DUMP IS DESIRED.  002600  COMAIDP   267   A
COMAIDP  *                LWA  =  LWA OF CORE TO BE DUMPED. (OPTIONAL)  002610  COMAIDP   268   A
COMAIDP  *                LNG  =  NR OF WORDS TO BE DUMPED. *LWA* AND *LNG* ARE  002620  COMAIDP   269   A
COMAIDP  *                          RELATED AS FOLLOWS--  DUMP IS FROM  002630  COMAIDP   270   A
COMAIDP  *                          FWA TO FWA+LNG UNLESS LNG IS NULL OR ZERO.  002640  COMAIDP   271   A
COMAIDP  *                          IN THAT CASE, DUMP IS FROM FWA TO LWA.  002650  COMAIDP   272   A
COMAIDP  *                RG   =  NULL IF ALL REGISTERS ARE TO BE DUMPED  002660  COMAIDP   273   A
COMAIDP  *                     =  A LIST OF THE REGISTERS (ENCLOSED IN PARENS)  002670  COMAIDP   274   A
COMAIDP  *                          DESIRED TO BE DUMPED            002680  COMAIDP   275   A
COMAIDP  *                     =  NON-REGISTER/NON-NULL IF NO REGISTERS TO BE  002690  COMAIDP   276   A
COMAIDP  *                          DUMPED                          002700  COMAIDP   277   A
COMAIDP  *                LL   =  LOWER LIMIT (DEFAULT=1)           002710  COMAIDP   278   A
COMAIDP  *                UL   =  UPPER LIMIT (DEFAULT=100)         002720  COMAIDP   279   A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMAIDP

```
COMAIDP  *                INC   =  INCREMENT (DEFAULT=1)                        002730   COMAIDP      280     A
COMAIDP  *                NUSE  =  USE BLOCK FLAG. IF *NUSE* IS NULL, THE PARA-  002740   COMAIDP      281     A
COMAIDP  *                           METER LIST EXISTS IN *USE DEBUG*, ELSE IS   002750   COMAIDP      282     A
COMAIDP  *                           ASSEMBLED AFTER *RJ*                        002760   COMAIDP      283     A
COMAIDP  *                UF    =  A STRING OF USER SNAP FLAGS PROCESSED BY      002770   COMAIDP      284     A
COMAIDP  *                           INSTALLER SUPPLIED USF= MACRO IF DESIRED.   002780   COMAIDP      285     A
COMAIDP  *                           USF= SHOULD GENERATE 1 WORD.                002790   COMAIDP      286     A
COMAIDP  *                                                                       002800   COMAIDP      287     A
COMAIDP  *         NOTES-- AN INDIRECT ADDRESSING SYNTAX IS PROVIDED FOR THE FWA,002810   COMAIDP      288     A
COMAIDP  *         LWA, AND LNG PARAMETERS. ANY OF THESE PARAMETERS PRECEDED BY  002820   COMAIDP      289     A
COMAIDP  *         AN ASTERICK (*) DENOTES ONE LEVEL OF INDIRECT ADDRESSING.     002830   COMAIDP      290     A
COMAIDP  *         N LEVELS OF INDIRECT ADDRESSING ARE POSSIBLE (N.LE.2**21-1)   002840   COMAIDP      291     A
COMAIDP  *         BY STACKING ASTERICKS. (E.G.  SNAP  **X1,***LWA)              002850   COMAIDP      292     A
COMAIDP  *                                                                       002860   COMAIDP      293     A
COMAIDP  *                A SINGLE REGISTER OR AN ADDRESS EXPRESSION CAN BE USED 002870   COMAIDP      294     A
COMAIDP  *         FOR THE FWA, LWA, OR LNG PARAMETER, BUT A MIX (E.G. SNAP X1+1)002880   COMAIDP      295     A
COMAIDP  *         IS NOT ALLOWED.                                               002890   COMAIDP      296     A
COMAIDP  *                                                                       002900   COMAIDP      297     A
COMAIDP  *         EXIT   NONE                                                   002910   COMAIDP      298     A
COMAIDP  *                                                                       002920   COMAIDP      299     A
COMAIDP  *         USES   NONE        (ALL REGISTERS ARE SAVED AND RESTORED)     002930   COMAIDP      300     A
COMAIDP  *                                                                       002940   COMAIDP      301     A
COMAIDP  *         CALLS  FRK=,RGR=,SNG=,SNP=,USF=(IF DEF)                       002950   COMAIDP      302     A
COMAIDP                                                                          002960   COMAIDP      303     A
COMAIDP                                                                          002970   COMAIDP      304     A
COMAIDP            PURGMAC SNAP                                                  002980   COMAIDP      305     A
COMAIDP                                                                          002990   COMAIDP      306     A
COMAIDP            MACRO  SNAP,NAME,FWA,LWA,LNG,RG,LL,UL,INC,NUSE,UF             003000   COMAIDP      307     A
COMAIDP            LOCAL  '?ID#005,'?ID#008                                      003010   COMAIDP      308     A
COMAIDP            LOCAL  RTN                                                    003020   COMAIDP      309     A
COMAIDP            LOCAL  APL                                                    003030   COMAIDP      310     A
COMAIDP  *                                                                       003040   COMAIDP      311     A
COMAIDP   +        RJ     =XSNP=                                                003050   COMAIDP      312     A
COMAIDP   -        VFD    30/APL                                                003060   COMAIDP      313     A
COMAIDP  *                                                                       003070   COMAIDP      314     A
COMAIDP   #NUSE    IFC    EQ,/NUSE//                                            003080   COMAIDP      315     A
COMAIDP            USE    DEBUG                                                 003090   COMAIDP      316     A
COMAIDP  *                                                                       003100   COMAIDP      317     A
COMAIDP   #NUSE    ELSE                                                          003110   COMAIDP      318     A
COMAIDP            EQ     RTN                                                   003120   COMAIDP      319     A
COMAIDP   #NUSE    ENDIF                                                         003130   COMAIDP      320     A
COMAIDP   APL      BSS    0                                                     003140   COMAIDP      321     A
COMAIDP            FRK=   (NAME),LL,UL,INC                                      003150   COMAIDP      322     A
COMAIDP            SNG=   (FWA)                                                 003160   COMAIDP      323     A
COMAIDP            SNG=   (LWA)                                                 003170   COMAIDP      324     A
COMAIDP            SNG=   (LNG)                                                 003180   COMAIDP      325     A
COMAIDP   '?ID#005 BSS 0                                                         003190   COMAIDP      326     A
COMAIDP            USF=   (UF)                                                   003200   COMAIDP      327     A
COMAIDP  *                                                                       003210   COMAIDP      328     A
COMAIDP   '?ID#006 IFNE *P,60D                                                   003220   COMAIDP      329     A
COMAIDP            IFNE *P,0,2                                                   003230   COMAIDP      330     A
COMAIDP            POS 0                                                         003240   COMAIDP      331     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMAIDP

```
COMAIDP   '?ID#007 SKIP                                                    003250   COMAIDP      332     A
COMAIDP  *                                                                 003260   COMAIDP      333     A
COMAIDP   '?ID#006 ENDIF                                                   003270   COMAIDP      334     A
COMAIDP   '?ID#008 BSS 0                                                   003280   COMAIDP      335     A
COMAIDP  *                                                                 003290   COMAIDP      336     A
COMAIDP            IFNE   '?ID#008-'?ID#005,1,2                            003300   COMAIDP      337     A
COMAIDP   '?ID#007 ENDIF                                                   003310   COMAIDP      338     A
COMAIDP   P        ERR    *USF= DID NOT GENERATE EXACTLY 60D BITS. "SEQUENCE"  003320   COMAIDP      339     A
COMAIDP  *                                                                 003330   COMAIDP      340     A
COMAIDP            RGR=   (RG)                                             003340   COMAIDP      341     A
COMAIDP  *                                                                 003350   COMAIDP      342     A
COMAIDP   #NUSE    IFC    EQ,/NUSE//                                       003360   COMAIDP      343     A
COMAIDP            USE    *                                                003370   COMAIDP      344     A
COMAIDP  *                                                                 003380   COMAIDP      345     A
COMAIDP   #NUSE    ELSE                                                    003390   COMAIDP      346     A
COMAIDP   RTN      BSS    0                                                003400   COMAIDP      347     A
COMAIDP   #NUSE    ENDIF                                                   003410   COMAIDP      348     A
COMAIDP  *                                                                 003420   COMAIDP      349     A
COMAIDP   SNAP     ENDM                                                    003430   COMAIDP      350     A
COMAIDP   SNG=     SPACE  4,8                                              003440   COMAIDP      351     A
COMAIDP  **        SNG= - GENERATE INDIRECT/ADDRESS FIELDS.                003450   COMAIDP      352     A
COMAIDP  *                                                                 003460   COMAIDP      353     A
COMAIDP  *                                                                 003470   COMAIDP      354     A
COMAIDP  *         GENERATES--                                             003480   COMAIDP      355     A
COMAIDP  *                                                                 003490   COMAIDP      356     A
COMAIDP  *         VFD    18/0                                             003500   COMAIDP      357     A
COMAIDP  *         VFD    21/NR OF LEVELS OF INDIRECT ADDRESSING (0=DIRECT)  003510   COMAIDP      358     A
COMAIDP  *         VFD    21/ADDRESS AT BEGINNING OF INDIRECT CHAIN        003520   COMAIDP      359     A
COMAIDP  *                                                                 003530   COMAIDP      360     A
COMAIDP  *         SNG=   (*ADDR)                                          003540   COMAIDP      361     A
COMAIDP  *                                                                 003550   COMAIDP      362     A
COMAIDP  *         ENTRY  *        =  A LEVEL OF INDIRECT ADDRESSING.      003560   COMAIDP      363     A
COMAIDP  *                              NO * MEANS DIRECT ADDRESSING.      003570   COMAIDP      364     A
COMAIDP  *                              E.G. *ADDR MEANS THE CONTENTS OF ADDR  003580   COMAIDP      365     A
COMAIDP  *                              CONTAIN AN ADDRESS THAT IS TO BE USED BY 003590   COMAIDP      366     A
COMAIDP  *                              SNAP UTILITY.                      003600   COMAIDP      367     A
COMAIDP  *                 ADDR     =  THE BEGINNING OF THE ADDRESS CHAIN. 003610   COMAIDP      368     A
COMAIDP                                                                    003620   COMAIDP      369     A
COMAIDP                                                                    003630   COMAIDP      370     A
COMAIDP            PURGMAC SNG=                                            003640   COMAIDP      371     A
COMAIDP                                                                    003650   COMAIDP      372     A
COMAIDP   SNG=     MACRO  ARG                                              003660   COMAIDP      373     A
COMAIDP   .9       SET    -1                                               003670   COMAIDP      374     A
COMAIDP   +        VFD    18/0                                             003680   COMAIDP      375     A
COMAIDP  *                                                                 003690   COMAIDP      376     A
COMAIDP   .D       DUP    -1                                               003700   COMAIDP      377     A
COMAIDP   .9       SET    .9+1                                             003710   COMAIDP      378     A
COMAIDP   .1       MICRO  .9+1,1,/ARG/                                     003720   COMAIDP      379     A
COMAIDP            IFC    NE,/".1"/*/,1                                    003730   COMAIDP      380     A
COMAIDP            STOPDUP                                                 003740   COMAIDP      381     A
COMAIDP   .D       ENDD                                                    003750   COMAIDP      382     A
COMAIDP  *                                                                 003760   COMAIDP      383     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMAIDP

```
COMAIDP   .2       MICRO  .9+1,,/ARG/                                  003770  COMAIDP   384  A
COMAIDP  *                                                             003780  COMAIDP   385  A
COMAIDP   #RG      IF     REG,".2"                                     003790  COMAIDP   386  A
COMAIDP            IFEQ   .9,0,1                                       003800  COMAIDP   387  A
COMAIDP   .9       SET    .9+1                                         003810  COMAIDP   388  A
COMAIDP   .3       MICRO  1,1,/".2"/                                   003820  COMAIDP   389  A
COMAIDP   .4       MICRO  2,1,/".2"/                                   003830  COMAIDP   390  A
COMAIDP   .2       MICRO  1,,/=XIDP=SV".3"+".4"/                       003840  COMAIDP   391  A
COMAIDP  *                                                             003850  COMAIDP   392  A
COMAIDP   #RG      ELSE                                                003860  COMAIDP   393  A
COMAIDP   .5       MICRO  1,2,/".2"/                                   003870  COMAIDP   394  A
COMAIDP            IFC    EQ,/".5"/=X/,1                               003880  COMAIDP   395  A
COMAIDP   .2       MICRO  3,,/".2"/                                    003890  COMAIDP   396  A
COMAIDP            IF     -DEF,".2",1                                  003900  COMAIDP   397  A
COMAIDP            SKIP   1                                            003910  COMAIDP   398  A
COMAIDP            IF     -ABS,".2",1                                  003920  COMAIDP   399  A
COMAIDP   .2       MICRO  1,,/=X".2"/                                  003930  COMAIDP   400  A
COMAIDP   #RG      ENDIF                                               003940  COMAIDP   401  A
COMAIDP  *                                                             003950  COMAIDP   402  A
COMAIDP            VFD    21/.9                                        003960  COMAIDP   403  A
COMAIDP            VFD    21/".2"                                      003970  COMAIDP   404  A
COMAIDP   SNG=     ENDM                                                003980  COMAIDP   405  A
COMAIDP   USF=     SPACE  4,8                                          003990  COMAIDP   406  A
COMAIDP  **        USF= - GENERATE USER FLAG PARAMETER CELL.           004000  COMAIDP   407  A
COMAIDP  *                                                             004010  COMAIDP   408  A
COMAIDP  *                                                             004020  COMAIDP   409  A
COMAIDP  *         THIS IS A DUMMY TO CREATE A ZERO WORD IF THE USER   004030  COMAIDP   410  A
COMAIDP  *         DOES NOT DESIRE OWN CODE.                           004040  COMAIDP   411  A
COMAIDP  *                                                             004050  COMAIDP   412  A
COMAIDP  *         USF=   USF                                          004060  COMAIDP   413  A
COMAIDP  *                                                             004070  COMAIDP   414  A
COMAIDP  *         ENTRY  USF  = ANY LEGAL STRING OF CHARACTERS        004080  COMAIDP   415  A
COMAIDP  *                                                             004090  COMAIDP   416  A
COMAIDP  *         EXIT   NONE                                         004100  COMAIDP   417  A
COMAIDP  *                                                             004110  COMAIDP   418  A
COMAIDP  *         USES   NONE                                         004120  COMAIDP   419  A
COMAIDP  *                                                             004130  COMAIDP   420  A
COMAIDP  *         CALLS  NONE                                         004140  COMAIDP   421  A
COMAIDP                                                                004150  COMAIDP   422  A
COMAIDP   .1       IF     -MAC,USF=                                    004160  COMAIDP   423  A
COMAIDP   USF=     MACRO  USF                                          004170  COMAIDP   424  A
COMAIDP            CON    0                                            004180  COMAIDP   425  A
COMAIDP   USF=     ENDM                                                004190  COMAIDP   426  A
COMAIDP   .1       ENDIF                                               004200  COMAIDP   427  A
COMAIDP   COMAIDP  SPACE  4,8                                          004210  COMAIDP   428  A
COMAIDP            BASE   *                                            004220  COMAIDP   429  A
COMAIDP   QUAL$    IF     -DEF,QUAL$                                   004230  COMAIDP   430  A
COMAIDP            QUAL   *                                            004240  COMAIDP   431  A
COMAIDP   QUAL$    ENDIF                                               004250  COMAIDP   432  A
COMAIDP   COMAIDP  ENDX                                                004260  COMAIDP   433  A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - COMAIDP

```
                IDENTIFIER                TOTAL    ACTIVE

                COMAIDP                     433       433

                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMAREG

COMAREG  *COMDECK COMAREG                                                    COMAREG        1     A
COMAREG            CTEXT  COMAREG - REPLACEMENT FOR R= PSEUDO INSTRUCTION.   COMAREG        2     A
COMAREG   COMAREG  SPACE  4                                                  COMAREG        3     A
COMAREG  ***       COMAREG - REPLACEMENT FOR R= PSEUDO INSTRUCTION.          COMAREG        4     A
COMAREG  *                                                                   COMAREG        5     A
COMAREG  *         R. H. GOODELL.     71/06/01.                              COMAREG        6     A
COMAREG  *                                                                   COMAREG        7     A
COMAREG  *         CONTROL DATA  PROPRIETARY PRODUCT.                        COMAREG        8     A
COMAREG  *         COPYRIGHT CONTROL DATA CORP. 1971.                        COMAREG        9           I
                -CCG*94
COMAREG  *         COMMENT COPYRIGHT CONTROL DATA SYSTEMS, INC. 1994.        CCG*94         9     A
COMAREG   COMAREG  SPACE  4,7                                                COMAREG       10     A
COMAREG  ***              THE FOLLOWING SET OF  MACROS,  OPDEFS,  AND  CPOPS COMAREG       11     A
COMAREG  *         DEFINE PSEUDO MACHINE INSTRUCTIONS FOR SETTING A REGISTER COMAREG       12     A
COMAREG  *         TO  A  VALUE INVOLVING REGISTERS AND/OR CONSTANTS.  THESE COMAREG       13     A
COMAREG  *         INSTRUCTIONS GENERATE NO CODE AT ALL OR A 15-BIT  MACHINE COMAREG       14     A
COMAREG  *         INSTRUCTION WHENEVER POSSIBLE DEPENDING ON THE EXPRESSION COMAREG       15     A
COMAREG  *         VALUE  AND WHETHER *B1=1* IS DEFINED, WITHOUT MOST OF THE COMAREG       16     A
COMAREG  *         DEFICIENCIES OF THE R=  PSEUDO  INSTRUCTION  IN  COMPASS. COMAREG       17     A
COMAREG  *         EXAMPLES --                                               COMAREG       18     A
COMAREG  *                                                                   COMAREG       19     A
COMAREG  *         COMPASS         R= FORM          NEW FORM        COMMENTS COMAREG       20     A
COMAREG  *                                                                   COMAREG       21     A
COMAREG  *         SX1   1         R=   X1,1        =X1   1                  COMAREG       22     A
COMAREG  *         SA1   EXP       R=   A1,EXP      =A1   EXP                COMAREG       23     A
COMAREG  *         SB4   B4+1      R=   B4,B4+1     =B4   B4+1      NOTE 1.  COMAREG       24     A
COMAREG  *         SA6   A6+EXP    R=   A6,A6+EXP   =A6   A6+EXP    NOTE 2.  COMAREG       25     A
COMAREG  *         SB.I  B.J       ---------        =B.I  B.J       NOTE 3.  COMAREG       26     A
COMAREG  *         BX6   X4        ---------        =X6   X4        BOOLEAN. COMAREG       27     A
COMAREG  *                                                                   COMAREG       28     A
COMAREG  *         NOTE 1.   R= GENERATES 30 BITS.  NEW IS 15 BITS IF B1=1.  COMAREG       29     A
COMAREG  *         NOTE 2.   R= GENERATES 30 BITS.  NEW IS 15 BITS IF B1=1   COMAREG       30     A
COMAREG  *                        AND (EXP) IS EVALUATABLE AND = +- 1.       COMAREG       31     A
COMAREG  *         NOTE 3.   R= CANNOT HANDLE SYMBOLIC REGISTERS.  NEW FORM  COMAREG       32     A
COMAREG  *                        GENERATES NO CODE IF  (I) = (J).           COMAREG       33     A
COMAREG  *         * NOTE * THIS COMDECK WILL DISAPPEAR WHEN COMPASS 4.0 ARRIVES. COMAREG  34     A
COMAREG                                                                      COMAREG       35     A
COMAREG                                                                      COMAREG       36     A
COMAREG            BASE   OCTAL                                              COMAREG       37     A
COMAREG   =A       SPACE  4,7                                                COMAREG       38     A
COMAREG  *         A-REGISTER SETS.                                          COMAREG       39     A
COMAREG                                                                      COMAREG       40     A
COMAREG                                                                      COMAREG       41     A
COMAREG   =A       CPOP   0,560,100         =AI                              COMAREG       42     A
COMAREG                                                                      COMAREG       43     A
COMAREG   =A,Q     OPDEF  I,Q               =AI    K                         COMAREG       44     A

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMAREG

```
COMAREG    '?REG#S2 A.I,(Q)                                                    COMAREG    45    A
COMAREG    ENDM                                                                COMAREG    46    A
COMAREG                                                                        COMAREG    47    A
COMAREG   =A,A     OPDEF  I,J              =AI    AJ                           COMAREG    48    A
COMAREG    IFNE I,J,1                                                          COMAREG    49    A
COMAREG    SA.I A.J                                                            COMAREG    50    A
COMAREG    ENDM                                                                COMAREG    51    A
COMAREG                                                                        COMAREG    52    A
COMAREG   =A,AQ    OPDEF  I,J,Q            =AI    AJ+K                         COMAREG    53    A
COMAREG    '?REG#S1 A.I,A.J,(Q)                                                COMAREG    54    A
COMAREG    ENDM                                                                COMAREG    55    A
COMAREG                                                                        COMAREG    56    A
COMAREG   =A,A+B   CPOP   0,540,132        =AI    AJ+BK                        COMAREG    57    A
COMAREG   =A,A-B   CPOP   0,550,132        =AI    AJ-BK                        COMAREG    58    A
COMAREG                                                                        COMAREG    59    A
COMAREG   =A,B     CPOP   0,560,120        =AI    BJ                           COMAREG    60    A
COMAREG                                                                        COMAREG    61    A
COMAREG   =A,BQ    OPDEF  I,J,Q            =AI    BJ+K                         COMAREG    62    A
COMAREG    '?REG#S1 A.I,B.J,(Q)                                                COMAREG    63    A
COMAREG    ENDM                                                                COMAREG    64    A
COMAREG                                                                        COMAREG    65    A
COMAREG   =A,B+A   CPOP   0,540,123        =AI    BK+AJ                        COMAREG    66    A
COMAREG   =A,B+B   CPOP   0,560,132        =AI    BJ+BK                        COMAREG    67    A
COMAREG   =A,B+X   CPOP   0,530,123        =AI    BK+XJ                        COMAREG    68    A
COMAREG   =A,B-B   CPOP   0,570,132        =AI    BJ-BK                        COMAREG    69    A
COMAREG   =A,-B    CPOP   0,570,102        =AI    -BK                          COMAREG    70    A
COMAREG   =A,-B+A  CPOP   0,550,123        =AI    -BK+AJ                       COMAREG    71    A
COMAREG   =A,-B+B  CPOP   0,570,123        =AI    -BK+BJ                       COMAREG    72    A
COMAREG                                                                        COMAREG    73    A
COMAREG   =A,X     CPOP   0,530,120        =AI    XJ                           COMAREG    74    A
COMAREG                                                                        COMAREG    75    A
COMAREG   =A,XQ    OPDEF  I,J,Q            =AI    XJ+K                         COMAREG    76    A
COMAREG    '?REG#S0 A.I,X.J,(Q)                                                COMAREG    77    A
COMAREG    ENDM                                                                COMAREG    78    A
COMAREG                                                                        COMAREG    79    A
COMAREG   =A,X+B   CPOP   0,530,132        =AI    XJ+BK                        COMAREG    80    A
COMAREG   =B       SPACE  4,7                                                  COMAREG    81    A
COMAREG  *         B-REGISTER SETS.                                            COMAREG    82    A
COMAREG                                                                        COMAREG    83    A
COMAREG                                                                        COMAREG    84    A
COMAREG   =B       CPOP   0,660,100        =BI                                 COMAREG    85    A
COMAREG                                                                        COMAREG    86    A
COMAREG   =B,Q     OPDEF  I,Q              =BI    K                            COMAREG    87    A
COMAREG    '?REG#S2 B.I,(Q)                                                    COMAREG    88    A
COMAREG    ENDM                                                                COMAREG    89    A
COMAREG                                                                        COMAREG    90    A
COMAREG   =B,A     CPOP   0,640,120        =BI    AJ                           COMAREG    91    A
COMAREG                                                                        COMAREG    92    A
COMAREG   =B,AQ    OPDEF  I,J,Q            =BI    AJ+K                         COMAREG    93    A
COMAREG    '?REG#S1 B.I,A.J,(Q)                                                COMAREG    94    A
COMAREG    ENDM                                                                COMAREG    95    A
COMAREG                                                                        COMAREG    96    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMAREG

```
COMAREG   =B,A+B   CPOP   0,640,132         =BI    AJ+BK                    COMAREG      97    A
COMAREG   =B,A-B   CPOP   0,650,132         =BI    AJ-BK                    COMAREG      98    A
COMAREG                                                                     COMAREG      99    A
COMAREG   =B,B     OPDEF  I,J               =BI    BJ                       COMAREG     100    A
COMAREG    IFNE I,J,1                                                       COMAREG     101    A
COMAREG    SB.I B.J                                                         COMAREG     102    A
COMAREG    ENDM                                                             COMAREG     103    A
COMAREG                                                                     COMAREG     104    A
COMAREG   =B,BQ    OPDEF  I,J,Q             =BI    BJ+K                     COMAREG     105    A
COMAREG    '?REG#S1 B.I,B.J,(Q)                                             COMAREG     106    A
COMAREG    ENDM                                                             COMAREG     107    A
COMAREG                                                                     COMAREG     108    A
COMAREG   =B,B+A   CPOP   0,640,123         =BI    BK+AJ                    COMAREG     109    A
COMAREG   =B,B+B   CPOP   0,660,132         =BI    BJ+BK                    COMAREG     110    A
COMAREG   =B,B-B   CPOP   0,670,132         =BI    BJ-BK                    COMAREG     111    A
COMAREG   =B,B+X   CPOP   0,630,123         =BI    BK+XJ                    COMAREG     112    A
COMAREG   =B,-B    CPOP   0,670,102         =BI    -BK                      COMAREG     113    A
COMAREG   =B,-B+A  CPOP   0,650,123         =BI    -BK+AJ                   COMAREG     114    A
COMAREG   =B,-B+B  CPOP   0,670,123         =BI    -BK+BJ                   COMAREG     115    A
COMAREG                                                                     COMAREG     116    A
COMAREG   =B,X     CPOP   0,630,120         =BI    XJ                       COMAREG     117    A
COMAREG                                                                     COMAREG     118    A
COMAREG   =B,XQ    OPDEF  I,J,Q             =BI    XJ+K                     COMAREG     119    A
COMAREG    '?REG#S0 B.I,X.J,(Q)                                             COMAREG     120    A
COMAREG    ENDM                                                             COMAREG     121    A
COMAREG                                                                     COMAREG     122    A
COMAREG   =B,X+B   CPOP   0,630,132         =BI    XJ+BK                    COMAREG     123    A
COMAREG   =X       SPACE  4,7                                               COMAREG     124    A
COMAREG  *         X-REGISTER SETS, PLUS SOME FAKES.                        COMAREG     125    A
COMAREG                                                                     COMAREG     126    A
COMAREG                                                                     COMAREG     127    A
COMAREG   =X       CPOP   0,130,111         =XI                BOOLEAN       COMAREG     128    A
COMAREG                                                                     COMAREG     129    A
COMAREG   =X,Q     OPDEF  I,Q               =XI    K                        COMAREG     130    A
COMAREG    '?REG#S2 X.I,(Q)                                                 COMAREG     131    A
COMAREG    ENDM                                                             COMAREG     132    A
COMAREG                                                                     COMAREG     133    A
COMAREG   =X,A     CPOP   0,740,120         =XI    AJ                       COMAREG     134    A
COMAREG                                                                     COMAREG     135    A
COMAREG   =X,AQ    OPDEF  I,J,Q             =XI    AJ+K                     COMAREG     136    A
COMAREG    '?REG#S1 X.I,A.J,(Q)                                             COMAREG     137    A
COMAREG    ENDM                                                             COMAREG     138    A
COMAREG                                                                     COMAREG     139    A
COMAREG   =X,A+B   CPOP   0,740,132         =XI    AJ+BK                    COMAREG     140    A
COMAREG   =X,A-B   CPOP   0,750,132         =XI    AJ-BK                    COMAREG     141    A
COMAREG                                                                     COMAREG     142    A
COMAREG   =X,B     CPOP   0,760,120         =XI    BJ                       COMAREG     143    A
COMAREG                                                                     COMAREG     144    A
COMAREG   =X,BQ    OPDEF  I,J,Q             =XI    BJ+K                     COMAREG     145    A
COMAREG    '?REG#S1 X.I,B.J,(Q)                                             COMAREG     146    A
COMAREG    ENDM                                                             COMAREG     147    A
COMAREG                                                                     COMAREG     148    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMAREG

```
COMAREG   =X,B+A   CPOP   0,740,123         =XI    BK+AJ                          COMAREG      149     A
COMAREG   =X,B+B   CPOP   0,760,132         =XI    BJ+BK                          COMAREG      150     A
COMAREG   =X,B-B   CPOP   0,770,132         =XI    BJ-BK                          COMAREG      151     A
COMAREG   =X,B+X   CPOP   0,730,123         =XI    BK+XJ                          COMAREG      152     A
COMAREG   =X,-B    CPOP   0,770,102         =XI    -BK                            COMAREG      153     A
COMAREG   =X,-B+A  CPOP   0,750,123         =XI    -BK+AJ                         COMAREG      154     A
COMAREG   =X,-B+B  CPOP   0,770,123         =XI    -BK+BJ                         COMAREG      155     A
COMAREG                                                                           COMAREG      156     A
COMAREG   =X,X     OPDEF  I,J               =XI    XJ          BOOLEAN            COMAREG      157     A
COMAREG    IFNE I,J,1                                                             COMAREG      158     A
COMAREG    BX.I X.J                                                               COMAREG      159     A
COMAREG    ENDM                                                                   COMAREG      160     A
COMAREG                                                                           COMAREG      161     A
COMAREG   =X,XQ    OPDEF  I,J,Q             =XI    XJ+K                           COMAREG      162     A
COMAREG    '?REG#S0 X.I,X.J,(Q)                                                   COMAREG      163     A
COMAREG    ENDM                                                                   COMAREG      164     A
COMAREG                                                                           COMAREG      165     A
COMAREG   =X,X+B   CPOP   0,730,132         =XI    XJ+BK                          COMAREG      166     A
COMAREG   =X,X+X   CPOP   0,360,132         =XI    XJ+XK       LONG ADD           COMAREG      167     A
COMAREG   =X,X-X   CPOP   0,370,132         =XI    XJ-XK       LONG ADD           COMAREG      168     A
COMAREG   =X,-X    CPOP   0,140,122         =XI    -XK         BOOLEAN            COMAREG      169     A
COMAREG   =X,-X+X  CPOP   0,370,123         =XI    -XK+XJ      LONG ADD           COMAREG      170     A
COMAREG   '?REG#S0 SPACE  4,7                                                     COMAREG      171     A
COMAREG  *         INTERNAL MACROS USED BY SOME OF THE ABOVE.                     COMAREG      172     A
COMAREG                                                                           COMAREG      173     A
COMAREG                                                                           COMAREG      174     A
COMAREG            PURGMAC '?REG#S0,'?REG#S1,'?REG#S2                             COMAREG      175     A
COMAREG                                                                           COMAREG      176     A
COMAREG   '?REG#S0 MACRO  I,J,Q       FOR VARIABLE FIELD SYNTAX = XQ              COMAREG      177     A
COMAREG    IF DEF,B1=1+Q,8D                                                       COMAREG      178     A
COMAREG  '?REG#QQ SET Q                                                           COMAREG      179     A
COMAREG    IFEQ '?REG#QQ,1,2                                                      COMAREG      180     A
COMAREG    S;A J+B1                                                               COMAREG      181     A
COMAREG    SKIP 5                                                                 COMAREG      182     A
COMAREG    IFEQ '?REG#QQ,,3                                                       COMAREG      183     A
COMAREG    IFNE I,J,3                                                             COMAREG      184     A
COMAREG    S;A J                                                                  COMAREG      185     A
COMAREG    SKIP 1                                                                 COMAREG      186     A
COMAREG    S;A J+Q                                                                COMAREG      187     A
COMAREG    ENDM                                                                   COMAREG      188     A
COMAREG                                                                           COMAREG      189     A
COMAREG   '?REG#S1 MACRO  I,J,Q       FOR VARIABLE FIELD SYNTAX = AQ OR BQ        COMAREG      190     A
COMAREG    IF DEF,B1=1+Q,11D                                                      COMAREG      191     A
COMAREG  '?REG#QQ SET Q                                                           COMAREG      192     A
COMAREG    IFEQ '?REG#QQ,1,2                                                      COMAREG      193     A
COMAREG    S;A J+B1                                                               COMAREG      194     A
COMAREG    SKIP 8D                                                                COMAREG      195     A
COMAREG    IFEQ '?REG#QQ,,3                                                       COMAREG      196     A
COMAREG    IFNE I,J,6                                                             COMAREG      197     A
COMAREG    S;A J                                                                  COMAREG      198     A
COMAREG    SKIP 4                                                                 COMAREG      199     A
COMAREG    IFEQ '?REG#QQ,-1,2                                                     COMAREG      200     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMAREG

```
COMAREG    S;A J-B1                                                          COMAREG      201     A
COMAREG    SKIP 1                                                            COMAREG      202     A
COMAREG    S;A J+Q                                                           COMAREG      203     A
COMAREG    ENDM                                                              COMAREG      204     A
COMAREG                                                                      COMAREG      205     A
COMAREG   '?REG#S2 MACRO  I,Q        FOR VARIABLE FIELD SYNTAX = Q           COMAREG      206     A
COMAREG    IF DEF,B1=1+Q,14D                                                 COMAREG      207     A
COMAREG  '?REG#QQ SET Q                                                      COMAREG      208     A
COMAREG    IFEQ '?REG#QQ,1,3                                                 COMAREG      209     A
COMAREG    IFNE I,B1,1                                                       COMAREG      210     A
COMAREG    S;A B1                                                            COMAREG      211     A
COMAREG    SKIP 10D                                                          COMAREG      212     A
COMAREG    IFEQ '?REG#QQ,,2                                                  COMAREG      213     A
COMAREG    S;A B0                                                            COMAREG      214     A
COMAREG    SKIP 7                                                            COMAREG      215     A
COMAREG    IFEQ '?REG#QQ,-1,2                                                COMAREG      216     A
COMAREG    S;A -B1                                                           COMAREG      217     A
COMAREG    SKIP 4                                                            COMAREG      218     A
COMAREG    IFEQ '?REG#QQ,2,2                                                 COMAREG      219     A
COMAREG    S;A B1+B1                                                         COMAREG      220     A
COMAREG    SKIP 1                                                            COMAREG      221     A
COMAREG    S;A Q                                                             COMAREG      222     A
COMAREG    ENDM                                                              COMAREG      223     A
COMAREG            SPACE  4,7                                                COMAREG      224     A
COMAREG            BASE   *                                                  COMAREG      225     A
COMAREG                                                                      COMAREG      226     A
COMAREG                                                                      COMAREG      227     A
COMAREG   COMAREG  ENDX                                                      COMAREG      228     A
```

SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - COMAREG

IDENTIFIER	TOTAL	ACTIVE
COMAREG	228	227
CCG*94	1	1

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

```
COMATOK  *COMDECK COMATOK                                                    COMATOK        1     A
COMATOK   COMATOK  CTEXT  COMATOK 2.0 - THE *TOGEL* MACROS.                  COMATOK        2     A
COMATOK   COMATOK  SPACE  4,10                                               COMATOK        3     A
COMATOK  ***       COMATOK - THE *TOGEL* MACROS.                             COMATOK        4     A
COMATOK  *         P CLARE   77/03/03                                        COMATOK        5     A
COMATOK  *         P CLARE   78/04/27 VERSION 2.0                            COMATOK        6     A
COMATOK   COMATOK  SPACE  4,10                                               COMATOK        7     A
COMATOK  ***       COMATOK - THE *TOGEL* MACROS.                             COMATOK        8     A
COMATOK  *                                                                   COMATOK        9     A
COMATOK  *                                                                   COMATOK       10     A
COMATOK  *         THIS COMDECK CONTAINS THE MACRO DEFINITIONS FOR *TOGEL*,  COMATOK       11     A
COMATOK  *         (TOKEN GENERATION LANGUAGE).  THESE MACROS ARE USED TO    COMATOK       12     A
COMATOK  *         GENERATE A *TOGEL OBJECT MODULE* (CALLED *TOM*) WHICH     COMATOK       13     A
COMATOK  *         CAN THEN BE FED TO *COMCTOK*, THE COMMON TOKEN GENERATOR. COMATOK       14     A
```

```
0        1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

```
COMATOK  *        *COMCTOK* WILL USE THE *TOM* GENERATED BY THESE MACROS TO          COMATOK  15  A
COMATOK  *        GENERATE TOKENS FOR THE HOST PROGRAM.                              COMATOK  16  A
COMATOK  *                                                                           COMATOK  17  A
COMATOK  *        *TOGEL* LOOKS SOMEWHAT LIKE A HIGHER LEVEL PROGRAMMING             COMATOK  18  A
COMATOK  *        LANGUAGE AND WAS DESIGNED SO AS TO ALLOW THE PROGRAMMER            COMATOK  19  A
COMATOK  *        TO CONTROL THE LOGIC OF TOKEN GENERATION AND AT THE SAME           COMATOK  20  A
COMATOK  *        TIME TO MINIMIZE THE EFFECTS OF LOW-LEVEL DETAILS.                 COMATOK  21  A
COMATOK  *                                                                           COMATOK  22  A
COMATOK  *        THE *TOGEL* MACROS ARE--                                           COMATOK  23  A
COMATOK  *                                                                           COMATOK  24  A
COMATOK  *              GROUP (A..Z),TOT,SQZ                                         COMATOK  25  A
COMATOK  *              GROUP (A..Z),TOT,NSQZ                                        COMATOK  26  A
COMATOK  *              IFT (A..Z)                                                   COMATOK  27  A
COMATOK  *                THEN                                                       COMATOK  28  A
COMATOK  *                ELST                                                       COMATOK  29  A
COMATOK  *                ENDT                                                       COMATOK  30  A
COMATOK  *              GOTO XX                                                      COMATOK  31  A
COMATOK  *              CALT XX                                                      COMATOK  32  A
COMATOK  *              CASEOF (A..Z)                                                COMATOK  33  A
COMATOK  *                TOKEN TOT,(TOT1,...,TOTN),PAD                              COMATOK  34  A
COMATOK  *                  .                                                        COMATOK  35  A
COMATOK  *                  .                                                        COMATOK  36  A
COMATOK  *                TOKEN TOT,(TOT1,...,TOTN),PAD                              COMATOK  37  A
COMATOK  *              ELSC                                                         COMATOK  38  A
COMATOK  *                TOKEN TOT,(TOT1,...,TOTN),PAD                              COMATOK  39  A
COMATOK  *              ENDC                                                         COMATOK  40  A
COMATOK  *                                                                           COMATOK  41  A
COMATOK  *        LITTLE OR NO EFFORT IS MADE TO DESCRIBE THE MECHANICS              COMATOK  42  A
COMATOK  *        OF USING THESE *TOGEL* MACROS HERE IN THE CODE.  THIS TOPIC        COMATOK  43  A
COMATOK  *        IS DEALT WITH THOROUGHLY IN THE FTN 5 IMS/DECK *LEX*.              COMATOK  44  A
COMATOK  *                                                                           COMATOK  45  A
COMATOK  *        IN THE FOLLOWING PARAGRAPHS, *ASSEMBLY-TIME DEPENDENCIES*          COMATOK  46  A
COMATOK  *        ARE INTERFACES THAT THE *COMATOK* INSTALLER MUST PROVIDE.          COMATOK  47  A
COMATOK  *        *ASSEMBLY-TIME OPTIONS* REFER TO INTERFACES THAT ARE OPTIONAL,     COMATOK  48  A
COMATOK  *        I.E. INTERFACES THAT HAVE A DEFAULT VALUE.                         COMATOK  49  A
COMATOK  *                                                                           COMATOK  50  A
COMATOK  ************************************************************************   COMATOK  51  A
COMATOK  *                                                                           COMATOK  52  A
COMATOK  *        ASSEMBLY-TIME DEPENDENCIES --                                      COMATOK  53  A
COMATOK  *                                                                           COMATOK  54  A
COMATOK  ************************************************************************   COMATOK  55  A
COMATOK  *                                                                           COMATOK  56  A
COMATOK  *        THE INSTALLER MUST PROVIDE *COMSTOK* WHICH CONTAINS                COMATOK  57  A
COMATOK  *        *COMATOK/COMCTOK* INTERFACE SYMBOL DEFINITIONS.                    COMATOK  58  A
COMATOK  *                                                                           COMATOK  59  A
COMATOK  ************************************************************************   COMATOK  60  A
COMATOK  *                                                                           COMATOK  61  A
COMATOK  *        ASSEMBLY-TIME OPTIONS --                                           COMATOK  62  A
COMATOK  *                                                                           COMATOK  63  A
COMATOK  ************************************************************************   COMATOK  64  A
COMATOK  *                                                                           COMATOK  65  A
COMATOK  *        ALL ASSEMBLY-TIME OPTIONS ARE SET UP IN *COMSTOK*.                 COMATOK  66  A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

```
COMATOK  *                                                                  COMATOK     67    A
COMATOK  ***********************************************************************  COMATOK     68    A
COMATOK   COMATOK  SPACE  4,10                                              COMATOK     69    A
COMATOK  **        *COMATOK* SYMBOL NAMING CONVENTIONS.                     COMATOK     70    A
COMATOK  *                                                                  COMATOK     71    A
COMATOK  *                                                                  COMATOK     72    A
COMATOK  *         BECAUSE THE AUTHOR FINDS *COMPASS* MACRO CODE VERY DIFFICULT  COMATOK     73    A
COMATOK  *         TO READ, A FEW SYMBOL NAMING CONVENTIONS WERE ADOPTED IN  COMATOK     74    A
COMATOK  *         THE INTEREST OF READABILITY.  THE INTENT BEHIND THIS     COMATOK     75    A
COMATOK  *         CONVENTION IS TO PASS AS MUCH SEMANTIC INFORMATION ABOUT  COMATOK     76    A
COMATOK  *         "HOW" A SYMBOL IS USED IN THE SYMBOL NAME ITSELF.  IN THIS  COMATOK     77    A
COMATOK  *         WAY, IT IS HOPED THAT THE READER WILL SORT-OF KNOW WHAT  COMATOK     78    A
COMATOK  *         TO EXPECT OF A SYMBOL BY MERELY LOOKING AT ITS NAME.  THIS  COMATOK     79    A
COMATOK  *         IS NOT A GREAT SYMBOL NAMING CONVENTION, BUT I HOPE IT MAKES  COMATOK     80    A
COMATOK  *         THIS CODE A LITTLE EASIER TO READ.                       COMATOK     81    A
COMATOK  *                                                                  COMATOK     82    A
COMATOK  *         SYMBOLS WITHIN *COMATOK* ARE OF THE FORM--               COMATOK     83    A
COMATOK  *                                                                  COMATOK     84    A
COMATOK  *               '?TKSXXX                                           COMATOK     85    A
COMATOK  *                                                                  COMATOK     86    A
COMATOK  *         WHERE *'?TK* IS A COMMON PREFIX THAT IS USED TO AVOID    COMATOK     87    A
COMATOK  *                     CONFLICTS WITH ANY HOST (THE HOST SHOULD AVOID  COMATOK     88    A
COMATOK  *                     SYMBOLS OF THIS FORM).                       COMATOK     89    A
COMATOK  *               *S*   IS A SYMBOL TYPE, AND                        COMATOK     90    A
COMATOK  *               *XXX* IS A UNIQUE SYMBOL NAME.                     COMATOK     91    A
COMATOK  *                                                                  COMATOK     92    A
COMATOK  *         SYMBOL TYPES *S* ARE--                                   COMATOK     93    A
COMATOK  *               C = MICRO NAME, I.E. A CHARACTER STRING.           COMATOK     94    A
COMATOK  *               F = FLAG, I.E. A SYMBOL THAT IS USED AS A LOGICAL  COMATOK     95    A
COMATOK  *                   FLAG IN THE WAY ALL PROGRAMMERS HAVE COME TO   COMATOK     96    A
COMATOK  *                   KNOW AND LOVE.                                 COMATOK     97    A
COMATOK  *               M = MACRO NAME.                                    COMATOK     98    A
COMATOK  *               R = REMOTE BLOCK NAME (RMT).                       COMATOK     99    A
COMATOK  *               S = SET SYMBOL, I.E. AN ABSOLUTE EVALUATABLE EXPRESSION  COMATOK    100    A
COMATOK  *                   VALUE.                                         COMATOK    101    A
COMATOK  *               # = A LABEL, I.E. EITHER AN *IF/ELSE/ENDIF* LABEL OR  COMATOK    102    A
COMATOK  *                   A GENERATED LABEL.                             COMATOK    103    A
COMATOK   CALT     SPACE  4,10                                              COMATOK    104    A
COMATOK  **        *CALT* TOGEL INSTRUCTION/MACRO.                          COMATOK    105    A
COMATOK  *                                                                  COMATOK    106    A
COMATOK  *                                                                  COMATOK    107    A
COMATOK  *         CALT   CAD                                               COMATOK    108    A
COMATOK  *                                                                  COMATOK    109    A
COMATOK  *         ENTRY  CAD = ADDR TO TRANSFER CONTROL TO.                COMATOK    110    A
COMATOK  *                                                                  COMATOK    111    A
COMATOK  *         EXIT   NONE                                              COMATOK    112    A
COMATOK  *                                                                  COMATOK    113    A
COMATOK  *         USES   NONE                                              COMATOK    114    A
COMATOK  *                                                                  COMATOK    115    A
COMATOK  *         CALLS  NONE                                              COMATOK    116    A
COMATOK                                                                     COMATOK    117    A
COMATOK            PURGMAC CALT                                             COMATOK    118    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

```
COMATOK                                                                  COMATOK      119     A
COMATOK   CALT     MACRO  A                                              COMATOK      120     A
COMATOK  *                                                               COMATOK      121     A
COMATOK  *         GENERATE *TOGEL* BINARY INSTRUCTION.                  COMATOK      122     A
COMATOK  *                                                               COMATOK      123     A
COMATOK   '?TKS001 SET 60-TG.CADP-TG.CADL                                COMATOK      124     A
COMATOK   + VFD '?TKS001/0,TG.CADL/A-FW.TOK,TG.TOFUL/=XTOK=CAL-FW.TOK    COMATOK      125     A
COMATOK  *                                                               COMATOK      126     A
COMATOK   CALT     ENDM                                                  COMATOK      127     A
COMATOK   CASEOF   SPACE  4,10                                           COMATOK      128     A
COMATOK  **        *CASEOF* TOGEL INSTRUCTION/MACRO.                     COMATOK      129     A
COMATOK  *                                                               COMATOK      130     A
COMATOK  *                                                               COMATOK      131     A
COMATOK  *         FORM --                                               COMATOK      132     A
COMATOK  *                                                               COMATOK      133     A
COMATOK  *         CASEOF A,B                                            COMATOK      134     A
COMATOK  *                                                               COMATOK      135     A
COMATOK  *         EXAMPLES --                                           COMATOK      136     A
COMATOK  *                                                               COMATOK      137     A
COMATOK  *         CASEOF (+../),TAD                                     COMATOK      138     A
COMATOK  *         CASEOF (A..Z)                                         COMATOK      139     A
COMATOK  *                                                               COMATOK      140     A
COMATOK  *         ENTRY  A = RANGE OF CHARACTERS IN THIS *CASEOF*.      COMATOK      141     A
COMATOK  *                    NOTE: THIS MACRO GENERATES A MICRO OF THIS COMATOK      142     A
COMATOK  *                    PARAMETER AND USES ONLY THE 1ST AND LAST   COMATOK      143     A
COMATOK  *                    CHARACTERS OF THE MICRO (*COA* AND *COZ*). COMATOK      144     A
COMATOK  *                B = OPTIONAL ADDR OF *TOKEN* CHARACTER MAP FOR THIS COMATOK      145     A
COMATOK  *                    *CASEOF*.  IF ABSENT, THEN 1ST *TOKEN* FOLLOWING COMATOK      146     A
COMATOK  *                    THIS *CASEOF* WILL AUTOMATICALLY BECOME THE 1ST COMATOK      147     A
COMATOK  *                    *TOKEN* ELEMENT IN THE CHAR MAP FOR THIS *CASEOF*. COMATOK      148     A
COMATOK  *                                                               COMATOK      149     A
COMATOK  *         EXIT   NONE                                           COMATOK      150     A
COMATOK  *                                                               COMATOK      151     A
COMATOK  *         USES   NONE                                           COMATOK      152     A
COMATOK  *                                                               COMATOK      153     A
COMATOK  *         CALLS  NONE                                           COMATOK      154     A
COMATOK                                                                  COMATOK      155     A
COMATOK                                                                  COMATOK      156     A
COMATOK            PURGMAC CASEOF                                        COMATOK      157     A
COMATOK                                                                  COMATOK      158     A
COMATOK   CASEOF   MACRO  A,BB                                           COMATOK      159     A
COMATOK    LOCAL '?TKL001                                                COMATOK      160     A
COMATOK  *                                                               COMATOK      161     A
COMATOK  *         SET UP --                                             COMATOK      162     A
COMATOK  *                                                               COMATOK      163     A
COMATOK  *         '?TKSCOA = *COA*, 1ST CHAR IN *CASEOF* RANGE (IN DPC). COMATOK      164     A
COMATOK  *         '?TKSCOZ = *COZ*, LAST CHAR IN *CASEOF* RANGE (IN RCS). COMATOK      165     A
COMATOK  *                                                               COMATOK      166     A
COMATOK   '?TKCS01 MICRO 1,, A                                           COMATOK      167     A
COMATOK   '?TKS001 MICCNT '?TKCS01                                       COMATOK      168     A
COMATOK  *                                                               COMATOK      169     A
COMATOK   '?TKCS02 MICRO '?TKS001,1, "'?TKCS01"                          COMATOK      170     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

```
COMATOK   '?TKCS01 MICRO 1,1, "'?TKCS01"                                  COMATOK    171    A
COMATOK  *                                                                COMATOK    172    A
COMATOK   '?TKSCOA SET 1R"'?TKCS01"                                       COMATOK    173    A
COMATOK   '?TKS001 SET 1R"'?TKCS02"+10000B-'?TKSCOA                       COMATOK    174    A
COMATOK   '?TKCS02 OCTMIC '?TKS001,2                                      COMATOK    175    A
COMATOK   '?TKSCOZ SET "'?TKCS02"B                                        COMATOK    176    A
COMATOK  *                                                                COMATOK    177    A
COMATOK  *         SET UP ADDR OF *TOKEN* CHARACTER MAP FOR THIS *CASEOF*.  COMATOK    178    A
COMATOK  *                                                                COMATOK    179    A
COMATOK   '?TK#001 IFC NE, BB                                             COMATOK    180    A
COMATOK   '?TKFCOF SET -1                                                 COMATOK    181    A
COMATOK   '?TKCTAD MICRO                                                  COMATOK    182    A
COMATOK   '?TKCS01 MICRO 1,, BB                                           COMATOK    183    A
COMATOK  *                                                                COMATOK    184    A
COMATOK   '?TK#001 ELSE                                                   COMATOK    185    A
COMATOK   '?TKFCOF SET 0                                                  COMATOK    186    A
COMATOK   '?TKCTAD MICRO 1,, '?TKL001                                     COMATOK    187    A
COMATOK   '?TKCS01 MICRO 1,, '?TKL001                                     COMATOK    188    A
COMATOK  *                                                                COMATOK    189    A
COMATOK   '?TK#001 ENDIF                                                  COMATOK    190    A
COMATOK  *                                                                COMATOK    191    A
COMATOK  *         GENERATE TOGEL BINARY INSTRUCTION.                     COMATOK    192    A
COMATOK  *                                                                COMATOK    193    A
COMATOK   + VFD TG.COAL/'?TKSCOA,TG.COZL/'?TKSCOZ,TG.TADL/"'?TKCS01"-FW.TOK  COMATOK    194    A
COMATOK    VFD TG.TOFUL/=XTOK=COF-FW.TOK                                  COMATOK    195    A
COMATOK  *                                                                COMATOK    196    A
COMATOK  *         CONVERT '?TKSCOA TO RCS SO THAT *ELSC/ENDC* CAN PERFORM  COMATOK    197    A
COMATOK  *         *CASEOF* RANGE CHECK.                                  COMATOK    198    A
COMATOK  *                                                                COMATOK    199    A
COMATOK   '?TKSCOA SET 77B                                                COMATOK    200    A
COMATOK  *                                                                COMATOK    201    A
COMATOK   CASEOF ENDM                                                     COMATOK    202    A
COMATOK   CHMIC    SPACE  4,20                                            COMATOK    203    A
COMATOK  ***       CHMIC - CREATE CHARACTER SHIFT MASK MICRO.             COMATOK    204    A
COMATOK  *                                                                COMATOK    205    A
COMATOK  * LOC     CHMIC  (LET1,LET2,...,LETN),BICH                       COMATOK    206    A
COMATOK  *                                                                COMATOK    207    A
COMATOK  *         ENTRY  LOC = NAME OF THE MICRO TO BE CREATED.          COMATOK    208    A
COMATOK  *                LET(I) = CHARACTERS TO GENERATE CHARACTER SHIFT  COMATOK    209    A
COMATOK  *                         MASK MICRO FOR.  CHARACTERS CAN BE SPECIFIED  COMATOK    210    A
COMATOK  *                         IN 1 OF 3 WAYS--                       COMATOK    211    A
COMATOK  *                           1. AS A SINGLE CHARACTER, AS IN      COMATOK    212    A
COMATOK  *                              *CHMIC (A,B,C)*.                  COMATOK    213    A
COMATOK  *                           2. AS A 2 OCDIT NUMBER, AS IN        COMATOK    214    A
COMATOK  *                              *CHMIC (01,02,03)*.               COMATOK    215    A
COMATOK  *                           3. AS ANY EVALUATABLE *COMPASS*      COMATOK    216    A
COMATOK  *                              EXPRESSION  (.GE. 3 CHARS), AS IN COMATOK    217    A
COMATOK  *                              *CHMIC (1RA,02D,03B,5-1)*.        COMATOK    218    A
COMATOK  *                         ANY OF THE ABOVE FORMS MAY BE USED, AND COMATOK    219    A
COMATOK  *                         IN ANY COMBINATION, AS IN              COMATOK    220    A
COMATOK  *                         *CHMIC (A,02,03B,1RD,10D-4)*.          COMATOK    221    A
COMATOK  *                                                                COMATOK    222    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

```
COMATOK  *                        NOTE-- *CHMIC* GENERATES BITS FOR CHARACTERS          COMATOK   223   A
COMATOK  *                        MODULO 60D.  THIS IMPLIES, THEREFORE THAT             COMATOK   224   A
COMATOK  *                        0=60D, OR 1R:=1R\.                                    COMATOK   225   A
COMATOK  *                BICH = BIAS CHARACTER.  SPECIFIES BINARY VALUE OF             COMATOK   226   A
COMATOK  *                       CHARACTER THAT IS TO OCCUPY BIT 59 OF MICRO            COMATOK   227   A
COMATOK  *                       TO GENERATE.  SEE *COMCBUB*.                           COMATOK   228   A
COMATOK  *                                                                              COMATOK   229   A
COMATOK  *         EXIT   LOC = MICRO CONTAINING THE 20-OCTAL DIGIT VALUE               COMATOK   230   A
COMATOK  *                          (WITH *B* SUFFIX PRESENT).                          COMATOK   231   A
COMATOK  *                                                                              COMATOK   232   A
COMATOK  *         GENERATES A MICRO WITH 1-BITS IN SPECIFIED CHARACTER                 COMATOK   233   A
COMATOK  *         POSITIONS.                                                           COMATOK   234   A
COMATOK                                                                                 COMATOK   235   A
COMATOK                                                                                 COMATOK   236   A
COMATOK            MACRO  CHMIC,LOC,LET,BICH                                            COMATOK   237   A
COMATOK  *                                                                              COMATOK   238   A
COMATOK  *         SET UP DEFAULT *BICH*.                                               COMATOK   239   A
COMATOK  *                                                                              COMATOK   240   A
COMATOK   '?CH#001 IFC    EQ,/BICH//,2                                                  COMATOK   241   A
COMATOK   '?CHSBCH SET    00B                                                           COMATOK   242   A
COMATOK   '?CH#001 ELSE                                                                 COMATOK   243   A
COMATOK   '?CHSBCH SET    BICH                                                          COMATOK   244   A
COMATOK   '?CH#001 ENDIF                                                                COMATOK   245   A
COMATOK  *                                                                              COMATOK   246   A
COMATOK   '?CHCCSM MICRO  1,60, 0000000000_0000000000_0000000000_0000000000_00000       COMATOK   247   A
COMATOK  ,00000_0000000000                                                              COMATOK   248   A
COMATOK  *                                                                              COMATOK   249   A
COMATOK            IRP    LET                                                           COMATOK   250   A
COMATOK  *                                                                              COMATOK   251   A
COMATOK  *         TRANSFORM *LET* INTO ITS BINARY VALUE.                               COMATOK   252   A
COMATOK  *                                                                              COMATOK   253   A
COMATOK   '?CHCLET MICRO  1,, LET                                                       COMATOK   254   A
COMATOK   '?CHLLET MICCNT '?CHCLET                                                      COMATOK   255   A
COMATOK  *                                                                              COMATOK   256   A
COMATOK   '?CH#003 IFNE   '?CHLLET,0                                                    COMATOK   257   A
COMATOK   '?CH#001 IFEQ   '?CHLLET,1                                                    COMATOK   258   A
COMATOK   '?CHSLET SET    1R_LET                                                        COMATOK   259   A
COMATOK  *                                                                              COMATOK   260   A
COMATOK   '?CH#001 ELSE                                                                 COMATOK   261   A
COMATOK   '?CH#002 IFEQ   '?CHLLET,2                                                    COMATOK   262   A
COMATOK   '?CHSLET SET    LET_B                                                         COMATOK   263   A
COMATOK  *                                                                              COMATOK   264   A
COMATOK   '?CH#002 ELSE                                                                 COMATOK   265   A
COMATOK   '?CHSLET SET    LET                                                           COMATOK   266   A
COMATOK   '?CH#002 ENDIF                                                                COMATOK   267   A
COMATOK   '?CH#001 ENDIF                                                                COMATOK   268   A
COMATOK  *                                                                              COMATOK   269   A
COMATOK  *         CONVERT CHAR IN *'?CHSLET* TO *ROTATED CHAR SET* VALUE.              COMATOK   270   A
COMATOK  *                                                                              COMATOK   271   A
COMATOK   '?CHSRCS SET    '?CHSLET+10000B-'?CHSBCH                                      COMATOK   272   A
COMATOK   '?CHCRCS OCTMIC '?CHSRCS,2                                                    COMATOK   273   A
COMATOK   '?CHSRCS SET    "'?CHCRCS"B                                                   COMATOK   274   A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

```
COMATOK            IFGE   '?CHSRCS,60D,2                                  COMATOK    275    A
COMATOK   '?CHCRCS DECMIC '?CHSRCS,1                                       COMATOK    276    A
COMATOK   '?CHSRCS SET    "'?CHCRCS"                                       COMATOK    277    A
COMATOK  *                                                                 COMATOK    278    A
COMATOK  *         EXTRACT TRAILING BITS.                                  COMATOK    279    A
COMATOK  *                                                                 COMATOK    280    A
COMATOK   '?CHCTBI MICRO  '?CHSRCS+2,,/"'?CHCCSM" /                        COMATOK    281    A
COMATOK  *                                                                 COMATOK    282    A
COMATOK  *         EXTRACT LEADING BITS.                                   COMATOK    283    A
COMATOK  *                                                                 COMATOK    284    A
COMATOK   '?CHCLBI MICRO                                                   COMATOK    285    A
COMATOK            IFNE   '?CHSRCS,0,1                                     COMATOK    286    A
COMATOK   '?CHCLBI MICRO  1,'?CHSRCS, "'?CHCCSM"                           COMATOK    287    A
COMATOK  *                                                                 COMATOK    288    A
COMATOK  *         MERGE IN NEW BIT.                                       COMATOK    289    A
COMATOK  *                                                                 COMATOK    290    A
COMATOK   '?CHCCSM MICRO  1,60, "'?CHCLBI"1"'?CHCTBI"                      COMATOK    291    A
COMATOK   '?CH#003 ENDIF                                                   COMATOK    292    A
COMATOK            IRP                                                     COMATOK    293    A
COMATOK  *                                                                 COMATOK    294    A
COMATOK  *         CONVERT BINARY TO OCTAL.                                COMATOK    295    A
COMATOK  *                                                                 COMATOK    296    A
COMATOK   LOC       MICRO                                                  COMATOK    297    A
COMATOK   '?CHSS01 SET                                                     COMATOK    298    A
COMATOK   '?CH#002  DUP   60D/3                                            COMATOK    299    A
COMATOK   '?CHSS01 SET    1+'?CHSS01                                       COMATOK    300    A
COMATOK   '?CHCS01 MICRO  3*'?CHSS01-2,1, "'?CHCCSM"                       COMATOK    301    A
COMATOK   '?CHCS02 MICRO  3*'?CHSS01-1,1, "'?CHCCSM"                       COMATOK    302    A
COMATOK   '?CHCS03 MICRO  3*'?CHSS01-0,1, "'?CHCCSM"                       COMATOK    303    A
COMATOK   '?CHCS03 OCTMIC "'?CHCS01"*4+"'?CHCS02"*2+"'?CHCS03",1           COMATOK    304    A
COMATOK   LOC      MICRO  1,, "LOC""'?CHCS03"                              COMATOK    305    A
COMATOK   '?CH#002 ENDD                                                    COMATOK    306    A
COMATOK  *                                                                 COMATOK    307    A
COMATOK   LOC      MICRO  1,, "LOC"B                                       COMATOK    308    A
COMATOK   CHMIC    ENDM                                                    COMATOK    309    A
COMATOK   ELSC     SPACE  4,10                                             COMATOK    310    A
COMATOK  **        *ELSC* TOGEL INSTRUCTION/MACRO.                         COMATOK    311    A
COMATOK  *                                                                 COMATOK    312    A
COMATOK  *                                                                 COMATOK    313    A
COMATOK  *         ENTRY  NONE                                             COMATOK    314    A
COMATOK  *                                                                 COMATOK    315    A
COMATOK  *         EXIT   NONE                                             COMATOK    316    A
COMATOK  *                                                                 COMATOK    317    A
COMATOK  *         USES   NONE                                             COMATOK    318    A
COMATOK  *                                                                 COMATOK    319    A
COMATOK  *         CALLS  NONE                                             COMATOK    320    A
COMATOK                                                                    COMATOK    321    A
COMATOK            PURGMAC ELSC                                            COMATOK    322    A
COMATOK                                                                    COMATOK    323    A
COMATOK   ELSC     MACRO                                                   COMATOK    324    A
COMATOK  *                                                                 COMATOK    325    A
COMATOK    IFNE '?TKSCOA,'?TKSCOZ,1                                        COMATOK    326    A
```

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

```
COMATOK   P ERR *CASEOF* RANGE AND COUNT OF *TOKEN*S DO NOT AGREE "SEQUENCE"   COMATOK   327   A
COMATOK  *                                                                     COMATOK   328   A
COMATOK  *         SET '?TKFCOF TO INDICATE THAT WE HAVE PROCESSED AN *ELSC*.  COMATOK   329   A
COMATOK  *                                                                     COMATOK   330   A
COMATOK   '?TKFCOF SET 1                                                       COMATOK   331   A
COMATOK  *                                                                     COMATOK   332   A
COMATOK   ELSC ENDM                                                            COMATOK   333   A
COMATOK   ELST     SPACE  4,10                                                 COMATOK   334   A
COMATOK  **        *ELST* TOGEL INSTRUCTION/MACRO.                             COMATOK   335   A
COMATOK  *                                                                     COMATOK   336   A
COMATOK  *                                                                     COMATOK   337   A
COMATOK  * L       ELST                                                        COMATOK   338   A
COMATOK  *                                                                     COMATOK   339   A
COMATOK  *         ENTRY  L = *IFT/ELST/ENDT* LABEL.                           COMATOK   340   A
COMATOK  *                                                                     COMATOK   341   A
COMATOK  *         EXIT   NONE                                                 COMATOK   342   A
COMATOK  *                                                                     COMATOK   343   A
COMATOK  *         USES   NONE                                                 COMATOK   344   A
COMATOK  *                                                                     COMATOK   345   A
COMATOK  *         CALLS  GOTO,'?TKMMEQ                                        COMATOK   346   A
COMATOK                                                                        COMATOK   347   A
COMATOK            PURGMAC ELST                                                COMATOK   348   A
COMATOK                                                                        COMATOK   349   A
COMATOK            MACRO  ELST,L                                               COMATOK   350   A
COMATOK  *                                                                     COMATOK   351   A
COMATOK  *         POP ENTRIES OFF '?TK1NNN UNTIL A MATCHING LABEL FOUND       COMATOK   352   A
COMATOK  *         OR UNTIL NO MORE ENTRIES.  THIS PROCESS IS SIMILAR TO       COMATOK   353   A
COMATOK  *         THE *ENDT* POPPING OF '?TK1NNN EXCEPT THAT WITH *ELST*      COMATOK   354   A
COMATOK  *         POPPING, '?TK1NNN IS NOT ALTERED SO THAT WHEN THE EXPECTED  COMATOK   355   A
COMATOK  *         *ENDT* DOES OCCUR THAT '?TK1NNN IS READY TO BE POPPED       COMATOK   356   A
COMATOK  *         PERMANENTLY.                                                COMATOK   357   A
COMATOK  *                                                                     COMATOK   358   A
COMATOK   '?TKSLEN SET '?TK1LEN                                                COMATOK   359   A
COMATOK  *                                                                     COMATOK   360   A
COMATOK   '?TK#001 DUP '?TKSLEN                                                COMATOK   361   A
COMATOK  *                                                                     COMATOK   362   A
COMATOK   '?TKCS01 DECMIC '?TKSLEN,3                                           COMATOK   363   A
COMATOK   '?TKSLEN SET '?TKSLEN-1                                              COMATOK   364   A
COMATOK  *                                                                     COMATOK   365   A
COMATOK   '?TKCS02 '?TKMMEQ '?TK1"'?TKCS01"                                    COMATOK   366   A
COMATOK  *                                                                     COMATOK   367   A
COMATOK  *         SET UP --                                                   COMATOK   368   A
COMATOK  *                                                                     COMATOK   369   A
COMATOK  *         '?TKCLNR - LATEST *IFT/ELST* LABEL NR.                      COMATOK   370   A
COMATOK  *         '?TKCLAB - LATEST *IFT/ELST* LABEL, IF PRESENT.             COMATOK   371   A
COMATOK  *                                                                     COMATOK   372   A
COMATOK  *         NOTE: TEMPORARY MICRO NAMES ARE NOT USED BECAUSE THESE      COMATOK   373   A
COMATOK  *         MICROS HAVE TO SURVIVE A *GOTO* MACRO REFERENCE THAT IS     COMATOK   374   A
COMATOK  *         GENERATED AS A RESULT OF *ELST*.  I.E. IT IS BAD PRACTICE   COMATOK   375   A
COMATOK  *         TO ASSUME THAT A TEMPORARY SYMBOL WILL SURVIVE A MACRO      COMATOK   376   A
COMATOK  *         REFERENCE.                                                  COMATOK   377   A
COMATOK  *                                                                     COMATOK   378   A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

```
COMATOK   '?TKCLNR MICRO 1,3, "'?TKCS02"                                  COMATOK      379     A
COMATOK   '?TKCLAB MICRO 4,, "'?TKCS02"                                   COMATOK      380     A
COMATOK  *                                                                COMATOK      381     A
COMATOK  *         OUTPUT APPROPRIATE '?TKINNN INTERNAL LABEL SO THAT     COMATOK      382     A
COMATOK  *         *IFT* CAN BRANCH TO *ELST* IF *IFT* CONDITION IS FALSE. COMATOK      383     A
COMATOK  *                                                                COMATOK      384     A
COMATOK    IF DEF,'?TKI"'?TKCLNR",1                                       COMATOK      385     A
COMATOK   P ERR THIS *ELST* PREVIOUSLY DEFINED.                           COMATOK      386     A
COMATOK  *                                                                COMATOK      387     A
COMATOK  *         OUTPUT A *GOTO '?TKENNN* TO SKIP *ELST* PORTION OF     COMATOK      388     A
COMATOK  *         *IFT/ELST/ENDT*.                                       COMATOK      389     A
COMATOK  *                                                                COMATOK      390     A
COMATOK    GOTO '?TKE"'?TKCLNR"                                           COMATOK      391     A
COMATOK  *                                                                COMATOK      392     A
COMATOK   '?TKI"'?TKCLNR" BSS 0                                           COMATOK      393     A
COMATOK  *                                                                COMATOK      394     A
COMATOK  *         IF IN *TEST* MODE, OUTPUT *ELST* BINARY INSTRUCTION.   COMATOK      395     A
COMATOK  *                                                                COMATOK      396     A
COMATOK    IFNE TEST,0,1                                                  COMATOK      397     A
COMATOK   + VFD 42/0L"'?TKCLAB",TG.TOFUL/=XTOK=ELS-FW.TOK                 COMATOK      398     A
COMATOK  *                                                                COMATOK      399     A
COMATOK  *         CHECK *ELST* LABEL WITH '?TK1NNN LABEL.                COMATOK      400     A
COMATOK  *                                                                COMATOK      401     A
COMATOK    IFC EQ, "'?TKCLAB" L ,1                                        COMATOK      402     A
COMATOK    STOPDUP                                                        COMATOK      403     A
COMATOK  *                                                                COMATOK      404     A
COMATOK   '?TK#001 ENDD                                                   COMATOK      405     A
COMATOK  *                                                                COMATOK      406     A
COMATOK   ELST ENDM                                                       COMATOK      407     A
COMATOK   ENDC     SPACE  4,10                                            COMATOK      408     A
COMATOK  **        *ENDC* TOGEL INSTRUCTION/MACRO.                        COMATOK      409     A
COMATOK  *                                                                COMATOK      410     A
COMATOK  *                                                                COMATOK      411     A
COMATOK  *         ENTRY  NONE                                            COMATOK      412     A
COMATOK  *                                                                COMATOK      413     A
COMATOK  *         EXIT   NONE                                            COMATOK      414     A
COMATOK  *                                                                COMATOK      415     A
COMATOK  *         USES   NONE                                            COMATOK      416     A
COMATOK  *                                                                COMATOK      417     A
COMATOK  *         CALLS  NONE                                            COMATOK      418     A
COMATOK                                                                   COMATOK      419     A
COMATOK            PURGMAC ENDC                                           COMATOK      420     A
COMATOK                                                                   COMATOK      421     A
COMATOK   ENDC     MACRO                                                  COMATOK      422     A
COMATOK  *                                                                COMATOK      423     A
COMATOK     IFNE '?TKSCOA,'?TKSCOZ,1                                      COMATOK      424     A
COMATOK   P ERR *CASEOF* RANGE AND COUNT OF *TOKEN*S DO NOT AGREE "SEQUENCE" COMATOK      425     A
COMATOK  *                                                                COMATOK      426     A
COMATOK  *         REINITIALIZE '?TKFCOF.                                 COMATOK      427     A
COMATOK  *                                                                COMATOK      428     A
COMATOK   '?TKFCOF SET -1                                                 COMATOK      429     A
COMATOK  *                                                                COMATOK      430     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

```
COMATOK   ENDC ENDM                                                          COMATOK     431    A
COMATOK   ENDT     SPACE  4,10                                               COMATOK     432    A
COMATOK  **        *ENDT* TOGEL INSTRUCTION/MACRO.                           COMATOK     433    A
COMATOK  *                                                                   COMATOK     434    A
COMATOK  *                                                                   COMATOK     435    A
COMATOK  * L       ENDT                                                      COMATOK     436    A
COMATOK  *                                                                   COMATOK     437    A
COMATOK  *         ENTRY  L = *IFT/ELST/ENDT* LABEL.                         COMATOK     438    A
COMATOK  *                                                                   COMATOK     439    A
COMATOK  *         EXIT   NONE                                               COMATOK     440    A
COMATOK  *                                                                   COMATOK     441    A
COMATOK  *         CALLS  '?TKMMEQ                                           COMATOK     442    A
COMATOK                                                                      COMATOK     443    A
COMATOK            PURGMAC ENDT                                              COMATOK     444    A
COMATOK                                                                      COMATOK     445    A
COMATOK            MACRO  ENDT,L                                             COMATOK     446    A
COMATOK  *                                                                   COMATOK     447    A
COMATOK  *         POP ENTRIES OFF '?TK1NNN UNTIL A MATCHING LABEL FOUND     COMATOK     448    A
COMATOK  *         OR UNTIL NO MORE ENTRIES.                                 COMATOK     449    A
COMATOK  *                                                                   COMATOK     450    A
COMATOK   '?TK#001 DUP '?TK1LEN                                              COMATOK     451    A
COMATOK  *                                                                   COMATOK     452    A
COMATOK   '?TKCS01 DECMIC '?TK1LEN,3                                         COMATOK     453    A
COMATOK   '?TK1LEN SET '?TK1LEN-1                                            COMATOK     454    A
COMATOK   '?TKCS02 '?TKMMEQ '?TK1"'?TKCS01"                                  COMATOK     455    A
COMATOK  *                                                                   COMATOK     456    A
COMATOK  *         SET UP --                                                 COMATOK     457    A
COMATOK  *                                                                   COMATOK     458    A
COMATOK  *         '?TKCS01 = LATEST *IFT/ELST/ENDT* LABEL NR.               COMATOK     459    A
COMATOK  *         '?TKCS02 = LATEST *IFT/ELST/ENDT* LABEL, IF PRESENT.      COMATOK     460    A
COMATOK  *                                                                   COMATOK     461    A
COMATOK   '?TKCS01 MICRO 1,3, "'?TKCS02"                                     COMATOK     462    A
COMATOK   '?TKCS02 MICRO 4,, "'?TKCS02"                                      COMATOK     463    A
COMATOK  *                                                                   COMATOK     464    A
COMATOK  *         IF IN *TEST* MODE, OUTPUT *ENDT* BINARY INSTRUCTION.      COMATOK     465    A
COMATOK  *                                                                   COMATOK     466    A
COMATOK    IFNE TEST,0,1                                                     COMATOK     467    A
COMATOK  + VFD TG.IFLL/0L"'?TKCS02",TG.TOFUL/=XTOK=ENT-FW.TOK                COMATOK     468    A
COMATOK  *                                                                   COMATOK     469    A
COMATOK  *         OUTPUT APPROPRIATE INTERNAL LABELS FOR THIS '?TK1NNN ENTRY.  COMATOK     470    A
COMATOK  *                                                                   COMATOK     471    A
COMATOK    IF -DEF,'?TKI"'?TKCS01",1                                         COMATOK     472    A
COMATOK   '?TKI"'?TKCS01" BSS 0                                              COMATOK     473    A
COMATOK  *                                                                   COMATOK     474    A
COMATOK   '?TKE"'?TKCS01" BSS 0                                              COMATOK     475    A
COMATOK  *                                                                   COMATOK     476    A
COMATOK  *         CHECK *ENDT* LABEL WITH '?TK1NNN LABEL.                   COMATOK     477    A
COMATOK  *                                                                   COMATOK     478    A
COMATOK    IFC EQ, "'?TKCS02" L ,1                                           COMATOK     479    A
COMATOK    STOPDUP                                                           COMATOK     480    A
COMATOK  *                                                                   COMATOK     481    A
COMATOK   '?TK#001 ENDD                                                      COMATOK     482    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

```
COMATOK  *                                                          COMATOK    483    A
COMATOK   ENDT ENDM                                                 COMATOK    484    A
COMATOK   GENTOK   SPACE  4,10                                      COMATOK    485    A
COMATOK  **        *GENTOK* TOGEL INSTRUCTION/MACRO.                COMATOK    486    A
COMATOK  *                                                          COMATOK    487    A
COMATOK  *                                                          COMATOK    488    A
COMATOK  *         GENTOK TOT                                       COMATOK    489    A
COMATOK  *                                                          COMATOK    490    A
COMATOK  *         ENTRY  TOT = TOKEN TYPE TO GENERATE.             COMATOK    491    A
COMATOK  *                                                          COMATOK    492    A
COMATOK  *         EXIT   NONE                                      COMATOK    493    A
COMATOK  *                                                          COMATOK    494    A
COMATOK  *         USES   NONE                                      COMATOK    495    A
COMATOK  *                                                          COMATOK    496    A
COMATOK  *         CALLS  NONE                                      COMATOK    497    A
COMATOK                                                             COMATOK    498    A
COMATOK                                                             COMATOK    499    A
COMATOK            PURGMAC GENTOK                                   COMATOK    500    A
COMATOK                                                             COMATOK    501    A
COMATOK   GENTOK   MACRO  TOT                                       COMATOK    502    A
COMATOK  *                                                          COMATOK    503    A
COMATOK  *         GENERATE *TOGEL* BINARY INSTRUCTION.             COMATOK    504    A
COMATOK  *                                                          COMATOK    505    A
COMATOK   '?TKS001 SET 60-TG.GTTP-TG.GTTL                           COMATOK    506    A
COMATOK   + VFD '?TKS001/0,TG.GTTL/"O."TOT,TG.TOFUL/=XTOK=GEN       COMATOK    507    A
COMATOK   GENTOK   ENDM                                             COMATOK    508    A
COMATOK   GOTO     SPACE  4,10                                      COMATOK    509    A
COMATOK  **        *GOTO XX* TOGEL INSTRUCTION/MACRO.               COMATOK    510    A
COMATOK  *                                                          COMATOK    511    A
COMATOK  *         GOTO LAB                                         COMATOK    512    A
COMATOK  *                                                          COMATOK    513    A
COMATOK  *         ENTRY  LAB = *COMPASS* LOCATION FIELD TO *GOTO* TO.  COMATOK   514    A
COMATOK  *                                                          COMATOK    515    A
COMATOK  *         EXIT   NONE                                      COMATOK    516    A
COMATOK  *                                                          COMATOK    517    A
COMATOK  *         USES   NONE                                      COMATOK    518    A
COMATOK  *                                                          COMATOK    519    A
COMATOK  *         CALLS  NONE                                      COMATOK    520    A
COMATOK                                                             COMATOK    521    A
COMATOK                                                             COMATOK    522    A
COMATOK            PURGMAC GOTO                                     COMATOK    523    A
COMATOK                                                             COMATOK    524    A
COMATOK   GOTO     MACRO  A                                         COMATOK    525    A
COMATOK  *                                                          COMATOK    526    A
COMATOK  *         GENERATE *TOGEL* BINARY INSTRUCTION.             COMATOK    527    A
COMATOK  *                                                          COMATOK    528    A
COMATOK   '?TKS001 SET 60-TG.GOTP-TG.GOTL                           COMATOK    529    A
COMATOK   + VFD '?TKS001/0,TG.GOTL/A-FW.TOK,TG.TOFUL/=XTOK=GO-FW.TOK  COMATOK  530    A
COMATOK   GOTO ENDM                                                 COMATOK    531    A
COMATOK   GROUP    SPACE  4,10                                      COMATOK    532    A
COMATOK  **        *GROUP (A..Z),TOT,SQZ/NSQZ* TOGEL INSTRUCTION/MACRO.  COMATOK  533    A
COMATOK  *                                                          COMATOK    534    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

```
COMATOK  *                                                          COMATOK    535    A
COMATOK  *         GROUP  RNG,TOT,MODE                              COMATOK    536    A
COMATOK  *                                                          COMATOK    537    A
COMATOK  *         ENTRY  RNG  = A CHARACTER RANGE THAT SPECIFIES THE CHARACTERS  COMATOK    538    A
COMATOK  *                       THAT ARE TO BE GROUPED TOGETHER BY THIS *GROUP*.  COMATOK    539    A
COMATOK  *                       A CHARACTER RANGE CAN BE SPECIFIED VIA ANY  COMATOK    540    A
COMATOK  *                       COMBINATION OF THE FOLLOWING SYNTAXES--  COMATOK    541    A
COMATOK  *                         1. A CHARACTER RANGE CAN BE SPECIFIED VIA A  COMATOK    542    A
COMATOK  *                            SIMPLE LIST, AS IN *GROUP (ABCDEFG)*.  COMATOK    543    A
COMATOK  *                         2. A CHARACTER RANGE MAY BE NOTATED USING THE  COMATOK    544    A
COMATOK  *                            SPECIAL OPERATOR *..*, AS IN *GROUP (A..G)*  COMATOK    545    A
COMATOK  *                            WHICH READS "FROM A TO G".  COMATOK    546    A
COMATOK  *                         3. INDIVIDUAL CHARACTERS CAN BE SPECIFIED AS  COMATOK    547    A
COMATOK  *                            2 DIGIT OCTAL NUMBERS, VIA THE SPECIAL  COMATOK    548    A
COMATOK  *                            FORM */NN*, WHERE */* IS A PREFIX OPERATOR,  COMATOK    549    A
COMATOK  *                            AND *NN* IS A 2 DIGIT OCTAL CONSTANT.  COMATOK    550    A
COMATOK  *                            E.G. *GROUP (/01../07)*.  COMATOK    551    A
COMATOK  *                                                          COMATOK    552    A
COMATOK  *                       EXAMPLES--                         COMATOK    553    A
COMATOK  *                         GROUP (A..Z0..9)                 COMATOK    554    A
COMATOK  *                         GROUP (+..:)                     COMATOK    555    A
COMATOK  *                         GROUP (+..*/77/00)               COMATOK    556    A
COMATOK  *                                                          COMATOK    557    A
COMATOK  *                TOT  = TOKEN TYPE TO ASSOCIATE WITH THIS *GROUP*.  COMATOK    558    A
COMATOK  *                       *COMATOK* WILL PREFIX THIS NAME WITH THE  COMATOK    559    A
COMATOK  *                       OPTIONALLY USER-DEFINED MICRO "O.".  COMATOK    560    A
COMATOK  *                                                          COMATOK    561    A
COMATOK  *                MODE = MODE OF *GROUP*ING.                COMATOK    562    A
COMATOK  *                     = *SQZ* MEANS THAT BLANKS ARE SQUEEZED (IGNORED).  COMATOK    563    A
COMATOK  *                     = *NSQZ* MEANS THAT BLANKS ARE NOT SQUEEZED.  COMATOK    564    A
COMATOK  *                                                          COMATOK    565    A
COMATOK  *         EXIT   NONE                                      COMATOK    566    A
COMATOK  *                                                          COMATOK    567    A
COMATOK  *         USES   NONE                                      COMATOK    568    A
COMATOK  *                                                          COMATOK    569    A
COMATOK  *         CALLS  '?TKMRNG                                  COMATOK    570    A
COMATOK                                                             COMATOK    571    A
COMATOK                                                             COMATOK    572    A
COMATOK            PURGMAC GROUP                                    COMATOK    573    A
COMATOK                                                             COMATOK    574    A
COMATOK   GROUP    MACRO  A,B,C                                     COMATOK    575    A
COMATOK            LOCAL  '?TK#001                                  COMATOK    576    A
COMATOK  *                                                          COMATOK    577    A
COMATOK  *         DETERMINE WHICH *TOFU* TO INVOKE.                COMATOK    578    A
COMATOK  *                                                          COMATOK    579    A
COMATOK   '?TKCTOF MICRO 1,, TOK=GS                                 COMATOK    580    A
COMATOK     IFC EQ, C NSQZ ,1                                       COMATOK    581    A
COMATOK   '?TKCTOF MICRO 1,, TOK=GN                                 COMATOK    582    A
COMATOK  *                                                          COMATOK    583    A
COMATOK  *         GENERATE *TOGEL* BINARY INSTRUCTION.             COMATOK    584    A
COMATOK  *                                                          COMATOK    585    A
COMATOK   '?TKS001 SET 60-TG.TOTP-TG.TOTL                           COMATOK    586    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

```
COMATOK   + VFD '?TKS001/0,TG.TOTL/"O."B,TG.MXAL/'?TK#001-FW.TOK,_________________  COMATOK  587  A
COMATOK  ,__TG.TOFUL/=X"'?TKCTOF"-FW.TOK                                            COMATOK  588  A
COMATOK  *                                                                          COMATOK  589  A
COMATOK  *         GENERATE CHARACTER RANGE SHIFT MASK.                             COMATOK  590  A
COMATOK  *                                                                          COMATOK  591  A
COMATOK     '?TKMRNG (A)                                                            COMATOK  592  A
COMATOK  *                                                                          COMATOK  593  A
COMATOK   '?TKRCSM '?TKMRMT ('?TK#001 LIT "'?TKCRNG")                               COMATOK  594  A
COMATOK  *                                                                          COMATOK  595  A
COMATOK   GROUP ENDM                                                                COMATOK  596  A
COMATOK   IFT      SPACE  4,10                                                      COMATOK  597  A
COMATOK  **        *IFT (A..Z)* TOGEL INSTRUCTION/MACRO.                            COMATOK  598  A
COMATOK  *                                                                          COMATOK  599  A
COMATOK  *                                                                          COMATOK  600  A
COMATOK  *         IFT RNG                                                          COMATOK  601  A
COMATOK  *                                                                          COMATOK  602  A
COMATOK  *         ENTRY  RNG = A CHARACTER RANGE THAT SPECIFIES THE CHARACTERS     COMATOK  603  A
COMATOK  *                      THAT ARE TO BE TESTED FOR.  SEE *GROUP* MACRO       COMATOK  604  A
COMATOK  *                      FOR A DESCRIPTION OF *RNG* SYNTAX.                  COMATOK  605  A
COMATOK  *                                                                          COMATOK  606  A
COMATOK  *         EXIT   NONE                                                      COMATOK  607  A
COMATOK  *                                                                          COMATOK  608  A
COMATOK  *         USES   NONE                                                      COMATOK  609  A
COMATOK  *                                                                          COMATOK  610  A
COMATOK  *         CALLS  '?TKMRNG                                                  COMATOK  611  A
COMATOK                                                                             COMATOK  612  A
COMATOK                                                                             COMATOK  613  A
COMATOK            PURGMAC IFT                                                      COMATOK  614  A
COMATOK                                                                             COMATOK  615  A
COMATOK    MACRO IFT,L,A                                                            COMATOK  616  A
COMATOK    LOCAL '?TK#001                                                           COMATOK  617  A
COMATOK  *                                                                          COMATOK  618  A
COMATOK  *         SET UP--                                                         COMATOK  619  A
COMATOK  *                                                                          COMATOK  620  A
COMATOK  *           '?TKSIGL   = GENERATED LABEL NR FOR *IFT/ELST/ENDT*            COMATOK  621  A
COMATOK  *                        IMPLIED JUMPS.  '?TKSIGL WILL BE USED TO          COMATOK  622  A
COMATOK  *                        GENERATE 2 UNIQUE LABELS: ONE FOR THE IMPLIED     COMATOK  623  A
COMATOK  *                        FALSE JUMP FROM THE *IFT* TO THE *ELST*, AND      COMATOK  624  A
COMATOK  *                        ONE FOR THE IMPLIED JUMP FROM THE *ELST* TO       COMATOK  625  A
COMATOK  *                        THE *ENDT*.                                       COMATOK  626  A
COMATOK  *                                                                          COMATOK  627  A
COMATOK  *                        *IFT/ELST* BRANCH IS TO A LABEL OF THE FORM --    COMATOK  628  A
COMATOK  *                                                                          COMATOK  629  A
COMATOK  *                             '?TKINNN                                     COMATOK  630  A
COMATOK  *                                                                          COMATOK  631  A
COMATOK  *                        WHERE *I* IS A SYMBOL TYPE MEANING *IFT/ELST*     COMATOK  632  A
COMATOK  *                        GENERATED LABEL, AND *NNN* IS A 3 DIGIT DECIMAL   COMATOK  633  A
COMATOK  *                        CONVERSION OF '?TKSIGL  (VIA *DECMIC*).           COMATOK  634  A
COMATOK  *                                                                          COMATOK  635  A
COMATOK  *                        *ELST/ENDT* BRANCH IS TO A LABEL OF THE FORM --   COMATOK  636  A
COMATOK  *                                                                          COMATOK  637  A
COMATOK  *                             '?TKENNN                                     COMATOK  638  A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

```
COMATOK  *                                                                  COMATOK     639    A
COMATOK  *                      WHERE *E* MEANS *ELST/ENDT* GENERATED LABEL.  COMATOK     640    A
COMATOK  *                                                                  COMATOK     641    A
COMATOK  *         FOR EXAMPLE, CONSIDER THE FOLLOWING --                   COMATOK     642    A
COMATOK  *                                                                  COMATOK     643    A
COMATOK  *                 IFT (A..Z) ---------------+ FALSE BRANCH        COMATOK     644    A
COMATOK  *                  .                        +                      COMATOK     645    A
COMATOK  *                 GOTO '?TKE001 (IMPLIED) --+-+                   COMATOK     646    A
COMATOK  *                 ELST                      + +                   COMATOK     647    A
COMATOK  *         '?TKI001 BSS 0   <-----------------+ +                  COMATOK     648    A
COMATOK  *                  .                          +                    COMATOK     649    A
COMATOK  *                 ENDT                        +                    COMATOK     650    A
COMATOK  *         '?TKE001 BSS 0   <-------------------+                   COMATOK     651    A
COMATOK  *                                                                  COMATOK     652    A
COMATOK   '?TKSIGL SET '?TKSIGL+1                                           COMATOK     653    A
COMATOK    IFGT '?TKSIGL,999D,1                                             COMATOK     654    A
COMATOK   P ERR TOO MANY *IFT/THEN/ELST/ENDT* BLOCKS. "SEQUENCE"            COMATOK     655    A
COMATOK  *                                                                  COMATOK     656    A
COMATOK   '?TKCIGL DECMIC '?TKSIGL,3                                        COMATOK     657    A
COMATOK  *                                                                  COMATOK     658    A
COMATOK  *         GENERATE *TOGEL* BINARY INSTRUCTION.                     COMATOK     659    A
COMATOK  *                                                                  COMATOK     660    A
COMATOK   '?TKS001 SET 60-TG.FALP-TG.FALL                                   COMATOK     661    A
COMATOK   + VFD '?TKS001/0,TG.FALL/'?TKI"'?TKCIGL"-FW.TOK,_______________   COMATOK     662    A
COMATOK  ,__TG.MXAL/'?TK#001-FW.TOK,TG.TOFUL/=XTOK=IFT-FW.TOK               COMATOK     663    A
COMATOK  *                                                                  COMATOK     664    A
COMATOK  *         GENERATE CHAR RANGE SHIFT MASK.                          COMATOK     665    A
COMATOK  *                                                                  COMATOK     666    A
COMATOK    '?TKMRNG (A)                                                     COMATOK     667    A
COMATOK   '?TKRCSM '?TKMRMT ('?TK#001 LIT "'?TKCRNG")                       COMATOK     668    A
COMATOK  *                                                                  COMATOK     669    A
COMATOK  *         MAKE '?TK1NNN ENTRY FOR THIS LABEL.  '?TK1NNN IS A MICRO COMATOK     670    A
COMATOK  *         ARRAY THAT CONTAINS ALL THE CURRENTLY ACTIVE *IFT/ELST/ENDT* COMATOK     671    A
COMATOK  *         LABELS.  EACH ENTRY IN '?TK1NNN CONTAINS THE LABEL NAME  COMATOK     672    A
COMATOK  *         (AS SPECIFIED VIA *LAB* PARAMETER), THE *'?TKI* AND *'?TKE* COMATOK     673    A
COMATOK  *         SUFFIX FOR THIS LABEL, AND THE STATUS OF THE LABEL (EXPLAINED COMATOK     674    A
COMATOK  *         BELOW).  THIS MICRO ARRAY CAN BE THOUGHT OF AS A SYMBOL TABLE COMATOK     675    A
COMATOK  *         FOR THE ACTIVE *IFT/ELST/ENDT* LABELS AND IS USED AS A FORM COMATOK     676    A
COMATOK  *         OF COMMUNICATION BETWEEN THE *IFT/ELST/ENDT* MACROS SO THAT COMATOK     677    A
COMATOK  *         THEY CAN GENERATE THE APPROPRIATE LABELS AND IMPLIED *GOTO*S COMATOK     678    A
COMATOK  *         ASSOCIATED WITH THESE MACROS.                            COMATOK     679    A
COMATOK  *                                                                  COMATOK     680    A
COMATOK  *         EACH '?TK1NNN ENTRY IS A MICRO STRING OF THE FORM --     COMATOK     681    A
COMATOK  *                                                                  COMATOK     682    A
COMATOK  *               XXXLLL                                             COMATOK     683    A
COMATOK  *                                                                  COMATOK     684    A
COMATOK  *         WHERE *XXX* IS THE *'?TKSIGL* NR FOR THIS *IFT/ELST/ENDT* COMATOK     685    A
COMATOK  *                   BLOCK.                                         COMATOK     686    A
COMATOK  *              *S*  IS THE STATUS OF THE LABEL.                    COMATOK     687    A
COMATOK  *                   = *+* IF THIS BLOCK HAS BEEN DEFINED VIA *IFT*, COMATOK     688    A
COMATOK  *                   ELSE * * (BLANK(55B)).                         COMATOK     689    A
COMATOK  *              *LLL* IS THE LABEL ITSELF, AS SPECIFIED VIA *LAB*   COMATOK     690    A

         0        1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

```
COMATOK  *                    PARAMETER.  IF NO *LAB* WAS PRESENT, THEN *LLL*   COMATOK  691  A
COMATOK  *                    WILL BE NULL.                                     COMATOK  692  A
COMATOK  *                                                                      COMATOK  693  A
COMATOK   '?TK1LEN SET '?TK1LEN+1                                               COMATOK  694  A
COMATOK   '?TKCS01 DECMIC '?TK1LEN,3                                            COMATOK  695  A
COMATOK   '?TKCS02 MICRO 1,, '?TK1"'?TKCS01"                                    COMATOK  696  A
COMATOK  *                                                                      COMATOK  697  A
COMATOK   "'?TKCS02" MICRO 1,, "'?TKCIGL"_L                                     COMATOK  698  A
COMATOK  *                                                                      COMATOK  699  A
COMATOK   IFT ENDM                                                              COMATOK  700  A
COMATOK   ON       SPACE  4,10                                                  COMATOK  701  A
COMATOK  **        *ON I,V* TOGEL INSTRUCTION/MACRO.                            COMATOK  702  A
COMATOK  *                                                                      COMATOK  703  A
COMATOK  *                                                                      COMATOK  704  A
COMATOK  *         ON I,V                                                       COMATOK  705  A
COMATOK  *                                                                      COMATOK  706  A
COMATOK  *         ENTRY  I = INTERUPT TYPE.                                    COMATOK  707  A
COMATOK  *                V = VALUE TO STORE INTO INTERUPT FLAG.                COMATOK  708  A
COMATOK  *                                                                      COMATOK  709  A
COMATOK  *         EXIT   NONE                                                  COMATOK  710  A
COMATOK  *                                                                      COMATOK  711  A
COMATOK  *         USES   NONE                                                  COMATOK  712  A
COMATOK  *                                                                      COMATOK  713  A
COMATOK  *         CALLS  NONE                                                  COMATOK  714  A
COMATOK                                                                         COMATOK  715  A
COMATOK                                                                         COMATOK  716  A
COMATOK            PURGMAC ON                                                   COMATOK  717  A
COMATOK                                                                         COMATOK  718  A
COMATOK   ON       MACRO  I,V                                                   COMATOK  719  A
COMATOK  *                                                                      COMATOK  720  A
COMATOK   '?TKCS01 MICRO 1,4, I                                                 COMATOK  721  A
COMATOK   '?TKS001 SET 60-TG.INVP-TG.INVL                                       COMATOK  722  A
COMATOK   + VFD '?TKS001/0,TG.INVL/V,TG.INCL/TC."'?TKCS01",______________________ COMATOK  723  A
COMATOK  ,__TG.TOFUL/=XTOK=ON-FW.TOK                                            COMATOK  724  A
COMATOK  *                                                                      COMATOK  725  A
COMATOK   ON       ENDM                                                         COMATOK  726  A
COMATOK   PROC     SPACE  4,10                                                  COMATOK  727  A
COMATOK  **        *PROC* TOGEL INSTRUCTION/MACRO.                              COMATOK  728  A
COMATOK  *                                                                      COMATOK  729  A
COMATOK  *         DEFINE TOGEL *PROC* ENTRY POINT.  THIS MACRO IS MODELED      COMATOK  730  A
COMATOK  *         (I.E. COPIED) FROM *SUBR* MACRO IN *COMACPU*.                COMATOK  731  A
COMATOK  *                                                                      COMATOK  732  A
COMATOK  * PRN     PROC              DEFINE *EXIT.* AND *PRN_X*.                COMATOK  733  A
COMATOK  * PRN     PROC   0          SAME                                       COMATOK  734  A
COMATOK  * PRN     PROC   =          SAME, AND DEFINE *PRN* AS ENTRY POINT.     COMATOK  735  A
COMATOK  * PRN     PROC   -          DEFINE *EXIT.* BUT NOT *PRN_X*.            COMATOK  736  A
COMATOK  * PRN     PROC   LETTER     DEFINE *PRN_LETTER* BUT NOT *EXIT.*.       COMATOK  737  A
COMATOK  *                                                                      COMATOK  738  A
COMATOK  *         ENTRY  *PRN* = PROC NAME.                                    COMATOK  739  A
COMATOK  *                                                                      COMATOK  740  A
COMATOK  *         EXIT   TOGEL/COMPASS GENERATED --                            COMATOK  741  A
COMATOK  *                                                                      COMATOK  742  A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

```
COMATOK  *                PRN    GOTO  *+4S15   ALWAYS                                COMATOK   743   A
COMATOK  *                       ENTRY PRN      IF = SPECIFIED.                       COMATOK   744   A
COMATOK  *                EXIT.  SET   *        IF BLANK, 0, =, OR - SPECIFIED.       COMATOK   745   A
COMATOK  *                       NOREF EXIT.    IF BLANK, 0, =, OR - SPECIFIED.       COMATOK   746   A
COMATOK  *                PRN_X  EQU   *        IF BLANK, 0, OR = SPECIFIED.          COMATOK   747   A
COMATOK  *                       NOREF PRN_X    IF BLANK, 0, OR = SPECIFIED.          COMATOK   748   A
COMATOK  *                PRN_LETTER EQU *      IF LETTER SPECIFIED.                  COMATOK   749   A
COMATOK  *                                                                            COMATOK   750   A
COMATOK  *         USES   NONE                                                        COMATOK   751   A
COMATOK  *                                                                            COMATOK   752   A
COMATOK  *         CALLS NONE                                                         COMATOK   753   A
COMATOK                                                                               COMATOK   754   A
COMATOK                                                                               COMATOK   755   A
COMATOK            PURGMAC PROC                                                       COMATOK   756   A
COMATOK                                                                               COMATOK   757   A
COMATOK            MACRO  PROC,T,A                                                    COMATOK   758   A
COMATOK   T GOTO *+4S15D                                                              COMATOK   759   A
COMATOK   '?TKCS01 MICRO 1,,=T=                                                       COMATOK   760   A
COMATOK    IFC NE, A  ,3                                                              COMATOK   761   A
COMATOK    IFC LT, A 0 ,2                                                             COMATOK   762   A
COMATOK   "'?TKCS01" EQU *-1                                                          COMATOK   763   A
COMATOK    SKIP 7                                                                     COMATOK   764   A
COMATOK    IFC NE, A - ,4                                                             COMATOK   765   A
COMATOK    IFC EQ, A = ,1                                                             COMATOK   766   A
COMATOK    ENTRY T                                                                    COMATOK   767   A
COMATOK   "'?TKCS01"X EQU *-1                                                         COMATOK   768   A
COMATOK    NOREF "'?TKCS01"X                                                          COMATOK   769   A
COMATOK   EXIT. SET *-1                                                               COMATOK   770   A
COMATOK    NOREF EXIT.                                                                COMATOK   771   A
COMATOK   PROC ENDM                                                                   COMATOK   772   A
COMATOK   TOGEL    SPACE  4,10                                                        COMATOK   773   A
COMATOK  **        TOGEL - BEGIN/END *TOGEL* PROGRAM.                                 COMATOK   774   A
COMATOK  *                                                                            COMATOK   775   A
COMATOK  *                                                                            COMATOK   776   A
COMATOK  *         THIS MACRO IS USED TO DELIMIT A *TOGEL* PROGRAM.  THAT IS,         COMATOK   777   A
COMATOK  *         EVERY *TOGEL* PROGRAM SHOULD CONTAIN 2 *TOGEL* MACRO               COMATOK   778   A
COMATOK  *         REFERENCES: ONE AS THE 1ST LINE OF THE *TOGEL* PROGRAM,            COMATOK   779   A
COMATOK  *         AND ONE AS THE LAST LINE OF THE *TOGEL* PROGRAM.                   COMATOK   780   A
COMATOK  *                                                                            COMATOK   781   A
COMATOK  *         THIS MACRO SETS UP AND WINDS UP THE ASSEMBLY OF THE *TOGEL*        COMATOK   782   A
COMATOK  *         PROGRAM BY BOTH SETTING UP GLOBAL SYMBOLS AND BY FORCING           COMATOK   783   A
COMATOK  *         ASSEMBLY OF THE *RMT* BLOCKS.                                      COMATOK   784   A
COMATOK  *                                                                            COMATOK   785   A
COMATOK  *                          ** WARNING **                                     COMATOK   786   A
COMATOK  *                                                                            COMATOK   787   A
COMATOK  *         IF THE PROGRAMMER FORGETS A TERMINATING *TOGEL* REFERENCE,         COMATOK   788   A
COMATOK  *         ONLY PARTIAL ASSEMBLY OF THE *TOGEL* PROGRAM WILL OCCUR            COMATOK   789   A
COMATOK  *         (I.E. THE *RMT* BLOCKS ARE NOT ASSEMBLED), RESULTING IN            COMATOK   790   A
COMATOK  *         NUMEROUS ASSEMBLY ERRORS.                                          COMATOK   791   A
COMATOK  *                                                                            COMATOK   792   A
COMATOK  *                             * * *                                          COMATOK   793   A
COMATOK  *                                                                            COMATOK   794   A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

```
COMATOK  *         FORM --                                                    COMATOK    795    A
COMATOK  *                                                                    COMATOK    796    A
COMATOK  * PGM     TOGEL                                                      COMATOK    797    A
COMATOK  *                                                                    COMATOK    798    A
COMATOK  *         EXAMPLES --                                                COMATOK    799    A
COMATOK  *                                                                    COMATOK    800    A
COMATOK  * PGM1    TOGEL                                                      COMATOK    801    A
COMATOK  *           .                                                        COMATOK    802    A
COMATOK  *           .                                                        COMATOK    803    A
COMATOK  * PGM1    TOGEL                                                      COMATOK    804    A
COMATOK  *                                                                    COMATOK    805    A
COMATOK  * PGM2    TOGEL                                                      COMATOK    806    A
COMATOK  *           .                                                        COMATOK    807    A
COMATOK  *           .                                                        COMATOK    808    A
COMATOK  * PGM2    TOGEL                                                      COMATOK    809    A
COMATOK  *                                                                    COMATOK    810    A
COMATOK  *         ENTRY  PGM = OPTIONAL NAME OF *TOGEL* PROGRAM.  *TOGEL* USES COMATOK  811    A
COMATOK  *                      *PGM* TO VERIFY THAT THE BEGINNING AND END    COMATOK    812    A
COMATOK  *                      *TOGEL* MACRO REFS REFER TO THE SAME *TOGEL*  COMATOK    813    A
COMATOK  *                      PROGRAM.                                      COMATOK    814    A
COMATOK  *                                                                    COMATOK    815    A
COMATOK  *         EXIT   NONE                                                COMATOK    816    A
COMATOK  *                                                                    COMATOK    817    A
COMATOK  *         USES   NONE                                                COMATOK    818    A
COMATOK  *                                                                    COMATOK    819    A
COMATOK  *         CALLS  NONE                                                COMATOK    820    A
COMATOK                                                                       COMATOK    821    A
COMATOK                                                                       COMATOK    822    A
COMATOK            PURGMAC TOGEL                                              COMATOK    823    A
COMATOK                                                                       COMATOK    824    A
COMATOK            MACRO  TOGEL,A                                             COMATOK    825    A
COMATOK  *                                                                    COMATOK    826    A
COMATOK  *         CHECK TO SEE IF THIS IS A *HEADER* REF OR AN *END* REF     COMATOK    827    A
COMATOK  *         AND PROCESS ACCORDINGLY.                                   COMATOK    828    A
COMATOK  *                                                                    COMATOK    829    A
COMATOK  *         IN ADDITION, IF THIS IS A 1ST TIME EVER REF, THEN WE NEED  COMATOK    830    A
COMATOK  *         TO DEFINE A FEW SYMBOLS BEFORE PROCEEDING.                 COMATOK    831    A
COMATOK  *                                                                    COMATOK    832    A
COMATOK  *         '?TKFHDR = HEADER FLAG.                                    COMATOK    833    A
COMATOK  *                  = 0 IF THIS IS A *HEADER* REF (I.E. BEGINNING     COMATOK    834    A
COMATOK  *                      *TOGEL* PROGRAM), ELSE                        COMATOK    835    A
COMATOK  *                  = 1 IF THIS IS AN *END* REF (I.E. ENDING *TOGEL*  COMATOK    836    A
COMATOK  *                      PROGRAM).                                     COMATOK    837    A
COMATOK  *                                                                    COMATOK    838    A
COMATOK    IF -DEF,'?TKFHDR,2                                                 COMATOK    839    A
COMATOK  '?TKFHDR SET 0                                                       COMATOK    840    A
COMATOK  '?TKSHDR SET 0                                                       COMATOK    841    A
COMATOK  *                                                                    COMATOK    842    A
COMATOK   '?TK#001 IFEQ '?TKFHDR,0                                            COMATOK    843    A
COMATOK  *                                                                    COMATOK    844    A
COMATOK  *         HERE IF THIS IS A *HEADER* REF (I.E. BEGINNING *TOGEL* PGM). COMATOK  845    A
COMATOK  *         NEED TO --                                                 COMATOK    846    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

```
COMATOK  *          1. TOGGLE '?TKFHDR SO THAT NEXT *TOGEL* REF WILL BE *END*.      COMATOK    847    A
COMATOK  *          2. SET UP '?TKCHDR AS THE NAME OF THIS *TOGEL* PROGRAM.         COMATOK    848    A
COMATOK  *             IF *PGM* PARAMETER OCCURRED, THEN '?TKCHDR CONTAINS THE      COMATOK    849    A
COMATOK  *             SPECIFIED CHARACTERS IN PROGRAM NAME.  ELSE IF *PGM*         COMATOK    850    A
COMATOK  *             DID NOT OCCUR, THEN '?TKCHDR IS A STRING OF THE FORM--       COMATOK    851    A
COMATOK  *                '?TKHNNN                                                  COMATOK    852    A
COMATOK  *             WHERE *'?TKH* IS A PREFIX/SYMBOL TYPE, AND *NNN* IS          COMATOK    853    A
COMATOK  *             A 3 DIGIT DECIMAL NR THAT IS DERIVED FROM '?TKSHDR,          COMATOK    854    A
COMATOK  *             WHICH IS MERELY INCREMENTED FOR EACH NEW *TOGEL* PGM.        COMATOK    855    A
COMATOK  *                                                                          COMATOK    856    A
COMATOK   '?TKFHDR SET 1                                                            COMATOK    857    A
COMATOK   '?TKSHDR SET '?TKSHDR+1                                                   COMATOK    858    A
COMATOK  *                                                                          COMATOK    859    A
COMATOK   '?TKCHDR MICRO 1,, A                                                      COMATOK    860    A
COMATOK    IFC EQ, A  ,2                                                            COMATOK    861    A
COMATOK   '?TKCS01 DECMIC '?TKSHDR,3                                                COMATOK    862    A
COMATOK   '?TKCHDR MICRO 1,, '?TKH"'?TKCS01"                                        COMATOK    863    A
COMATOK  *                                                                          COMATOK    864    A
COMATOK  *         MISCELLANEOUS INITIALIZATIONS.                                   COMATOK    865    A
COMATOK  *                                                                          COMATOK    866    A
COMATOK    IF -DEF,'?TKSIGL,1                                                       COMATOK    867    A
COMATOK   '?TKSIGL SET 0                                                            COMATOK    868    A
COMATOK  *                                                                          COMATOK    869    A
COMATOK    IF -DEF,'?TKSCOF,1                                                       COMATOK    870    A
COMATOK   '?TKSCOF SET 0                                                            COMATOK    871    A
COMATOK  *                                                                          COMATOK    872    A
COMATOK    IF -DEF,'?TKSTLK,1                                                       COMATOK    873    A
COMATOK   '?TKSTLK SET 0                                                            COMATOK    874    A
COMATOK  *                                                                          COMATOK    875    A
COMATOK   '?TK1LEN SET 0                                                            COMATOK    876    A
COMATOK   '?TK2LEN SET 0                                                            COMATOK    877    A
COMATOK   '?TK3LEN SET 0                                                            COMATOK    878    A
COMATOK   '?TKFCOF SET -1                                                           COMATOK    879    A
COMATOK  *                                                                          COMATOK    880    A
COMATOK  *         HERE IF *END* REF (I.E. ENDING A *TOGEL* PROGRAM).               COMATOK    881    A
COMATOK  *         NEED TO --                                                       COMATOK    882    A
COMATOK  *           1. CHECK TO SEE THAT *PGM* NAME MATCHS '?TKCHDR.               COMATOK    883    A
COMATOK  *           2. FORCE ASSEMBLY OF *RMT* BLOCKS.                             COMATOK    884    A
COMATOK  *                                                                          COMATOK    885    A
COMATOK   '?TK#001 ELSE                                                             COMATOK    886    A
COMATOK   '?TKCS01 MICRO 1,, A                                                      COMATOK    887    A
COMATOK   '?TK#002 IFC NE, "'?TKCS01"                                               COMATOK    888    A
COMATOK    IFC NE, "'?TKCHDR" "'?TKCS01" ,1                                         COMATOK    889    A
COMATOK   '?TK#003 SKIP                                                             COMATOK    890    A
COMATOK   '?TK#002 ELSE                                                             COMATOK    891    A
COMATOK   '?TKCS02 MICRO 1,5, "'?TKCHDR"                                            COMATOK    892    A
COMATOK    IFC NE, "'?TKCHDR" '?TKH ,2                                              COMATOK    893    A
COMATOK   '?TK#003 ENDIF                                                            COMATOK    894    A
COMATOK   P ERR *TOGEL* PROGRAM DELIMITER NAMES DO NOT MATCH. "SEQUENCE"            COMATOK    895    A
COMATOK   '?TK#002 ENDIF                                                            COMATOK    896    A
COMATOK  *                                                                          COMATOK    897    A
COMATOK  *         SET ALL END-OF-FALSE CHAIN ENTRIES IN '?TK3NNN                   COMATOK    898    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

```
COMATOK  *         TO 0  (I.E. END-OF-CHAIN).                                  COMATOK    899    A
COMATOK  *                                                                     COMATOK    900    A
COMATOK   '?TKS001 SET 0                                                       COMATOK    901    A
COMATOK   '?TK#003 DUP '?TK3LEN                                                COMATOK    902    A
COMATOK   '?TKS001 SET '?TKS001+1                                              COMATOK    903    A
COMATOK  *                                                                     COMATOK    904    A
COMATOK   '?TKCS01 DECMIC '?TKS001,3                                           COMATOK    905    A
COMATOK   '?TKCS02 '?TKMMEQ '?TK3"'?TKCS01"                                    COMATOK    906    A
COMATOK   '?TKCS03 MICRO 1,3, "'?TKCS02"                                       COMATOK    907    A
COMATOK  *                                                                     COMATOK    908    A
COMATOK   '?TKT"'?TKCS03" = 0                                                  COMATOK    909    A
COMATOK   '?TK#003 ENDD                                                        COMATOK    910    A
COMATOK  *                                                                     COMATOK    911    A
COMATOK  *         FORCE *RMT* ASSEMBLIES.                                     COMATOK    912    A
COMATOK  *                                                                     COMATOK    913    A
COMATOK   '?TKRCSM HERE                                                        COMATOK    914    A
COMATOK   '?TKRTLK HERE                                                        COMATOK    915    A
COMATOK   '?TKRTOK HERE                                                        COMATOK    916    A
COMATOK   '?TKFHDR SET 0                                                       COMATOK    917    A
COMATOK   '?TK#001 ENDIF                                                       COMATOK    918    A
COMATOK   TOGEL ENDM                                                           COMATOK    919    A
COMATOK   TOKEN    SPACE  4,10                                                 COMATOK    920    A
COMATOK  **        *TOKEN* TOGEL INSTRUCTION/MACRO.                            COMATOK    921    A
COMATOK  *                                                                     COMATOK    922    A
COMATOK  *                                                                     COMATOK    923    A
COMATOK  * LAB     TOKEN  TOT,STX,PAD                                          COMATOK    924    A
COMATOK  *                                                                     COMATOK    925    A
COMATOK  *         ENTRY  LAB = LABEL FIELD FOR THIS *TOKEN* ENTRY.            COMATOK    926    A
COMATOK  *                TOT = TOKEN TYPE FOR THIS *TOKEN*.  CHARS IN *TOT*   COMATOK    927    A
COMATOK  *                      PARAMETER WILL BE PREFIXED WITH "O." MICRO.    COMATOK    928    A
COMATOK  *                STX = SYNTAX FOR THIS *TOKEN*.  USED TO DEFINE TOKENS COMATOK    929    A
COMATOK  *                      THAT ARE DEFINED VIA OTHER TOKENS.             COMATOK    930    A
COMATOK  *                PAD = PROCESSOR ADDR.  OPTIONAL PARAMETER THAT       COMATOK    931    A
COMATOK  *                      SPECIFIES A SPECIAL PROCESSOR THAT IS TO BE    COMATOK    932    A
COMATOK  *                      INVOKED WHEN AND IF THIS TOKEN IS GENERATED.   COMATOK    933    A
COMATOK  *                                                                     COMATOK    934    A
COMATOK  *         EXIT   NONE                                                 COMATOK    935    A
COMATOK  *                                                                     COMATOK    936    A
COMATOK  *         USES   NONE                                                 COMATOK    937    A
COMATOK  *                                                                     COMATOK    938    A
COMATOK  *         CALLS  '?TKMMEQ,'?TKMSMA,'?TKMTLK                           COMATOK    939    A
COMATOK                                                                        COMATOK    940    A
COMATOK            PURGMAC TOKEN                                               COMATOK    941    A
COMATOK                                                                        COMATOK    942    A
COMATOK            MACRO  TOKEN,A,BB,C,D                                       COMATOK    943    A
COMATOK  *                                                                     COMATOK    944    A
COMATOK   '?TK#001 IFC EQ, C                                                   COMATOK    945    A
COMATOK  *                                                                     COMATOK    946    A
COMATOK    IF -MIC,'?TKCTAD,1                                                  COMATOK    947    A
COMATOK   '?TKCTAD MICRO                                                       COMATOK    948    A
COMATOK   A '?TKMTOK BB,D,"'?TKCTAD",'?TKRTOK                                  COMATOK    949    A
COMATOK   '?TKCTAD MICRO                                                       COMATOK    950    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

```
COMATOK  *                                                                  COMATOK     951    A
COMATOK  *         IF '?TKFCOF .ZR., THEN NEED TO ADVANCE '?TKSCOA SO THAT  COMATOK     952    A
COMATOK  *         *ELSC/ENDC* CAN PERFORM *CASEOF* RANGE CHECK.            COMATOK     953    A
COMATOK  *                                                                  COMATOK     954    A
COMATOK    IFEQ '?TKFCOF,0,2                                                COMATOK     955    A
COMATOK   '?TKCS01 OCTMIC '?TKSCOA+1+10000B,2                               COMATOK     956    A
COMATOK   '?TKSCOA SET "'?TKCS01"B                                          COMATOK     957    A
COMATOK  *                                                                  COMATOK     958    A
COMATOK  *         HERE IF *STX* PARAMETER IS NON-NULL.                     COMATOK     959    A
COMATOK  *                                                                  COMATOK     960    A
COMATOK   '?TK#001 ELSE                                                     COMATOK     961    A
COMATOK   '?TK2LEN SET 0                                                    COMATOK     962    A
COMATOK  *                                                                  COMATOK     963    A
COMATOK   '?TK#002 IRP C                                                    COMATOK     964    A
COMATOK   '?TK2LEN SET '?TK2LEN+1                                           COMATOK     965    A
COMATOK   '?TKCS01 DECMIC '?TK2LEN,3                                        COMATOK     966    A
COMATOK  *                                                                  COMATOK     967    A
COMATOK   '?TK2"'?TKCS01" MICRO 1,, C                                       COMATOK     968    A
COMATOK  *                                                                  COMATOK     969    A
COMATOK   '?TK#002 IRP                                                      COMATOK     970    A
COMATOK  *                                                                  COMATOK     971    A
COMATOK  *         SEARCH FOR NAME OF SYMBOL THAT DEFINES NEXT *FALSE* TLINK COMATOK     972    A
COMATOK  *         ENTRY.                                                   COMATOK     973    A
COMATOK  *                                                                  COMATOK     974    A
COMATOK   '?TKCS01 '?TKMMEQ '?TK2"'?TKCS01"                                 COMATOK     975    A
COMATOK    '?TKMSMA 3,4                                                     COMATOK     976    A
COMATOK  *                                                                  COMATOK     977    A
COMATOK  *         SET UP SYMBOL NAME THAT IS TO BE THE LOCATION FIELD      COMATOK     978    A
COMATOK  *         OR THE BEGINNING OR THIS *TLINK* TRUE CHAIN.             COMATOK     979    A
COMATOK  *                                                                  COMATOK     980    A
COMATOK  *           1. IF A FIND IN '?TK3NNN, THEN THIS IS NOT THE 1ST *TLINK* COMATOK     981    A
COMATOK  *              ENTRY FOR THE CHAIN.  '?TK3NNN ENTRY CONTAINS       COMATOK     982    A
COMATOK  *              THE SYMBOL NAME FOR NEXT *FALSE* CHAIN ENTRY.       COMATOK     983    A
COMATOK  *                                                                  COMATOK     984    A
COMATOK  *           2. IF NO FIND IN '?TK3NNN, THEN THIS IS THE 1ST *TLINK* COMATOK     985    A
COMATOK  *              ENTRY FOR THE CHAIN.  SYMBOL NAME IS '?_TOT.  NEED  COMATOK     986    A
COMATOK  *              TO CREATE A '?TK3NNN ENTRY.                         COMATOK     987    A
COMATOK  *                                                                  COMATOK     988    A
COMATOK   '?TK#003 IFNE '?TKS001,0                                          COMATOK     989    A
COMATOK  *                                                                  COMATOK     990    A
COMATOK  *         HERE IF A FIND.                                          COMATOK     991    A
COMATOK  *                                                                  COMATOK     992    A
COMATOK   '?TKCS04 MICRO 1,3, "'?TKCS02"                                    COMATOK     993    A
COMATOK   '?TKCTLK MICRO 1,, '?TKT"'?TKCS04"                                COMATOK     994    A
COMATOK  *                                                                  COMATOK     995    A
COMATOK   '?TKSTLK SET '?TKSTLK+1                                           COMATOK     996    A
COMATOK    IFGT '?TKSTLK,999D,1                                             COMATOK     997    A
COMATOK   P ERR TOO MANY *TLINK* SYMBOLS "SEQUENCE"                         COMATOK     998    A
COMATOK  *                                                                  COMATOK     999    A
COMATOK   '?TKCS04 DECMIC '?TKSTLK,3                                        COMATOK    1000    A
COMATOK  *                                                                  COMATOK    1001    A
COMATOK  *         HERE IF NO FIND.                                         COMATOK    1002    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

```
COMATOK  *                                                                  COMATOK    1003    A
COMATOK   '?TK#003 ELSE                                                     COMATOK    1004    A
COMATOK  *                                                                  COMATOK    1005    A
COMATOK   '?TK3LEN SET '?TK3LEN+1                                           COMATOK    1006    A
COMATOK    IFGT '?TK3LEN,999D,1                                             COMATOK    1007    A
COMATOK   P ERR TOO MANY *TLINK* ENTRIES "SEQUENCE"                         COMATOK    1008    A
COMATOK  *                                                                  COMATOK    1009    A
COMATOK   '?TKSTLK SET '?TKSTLK+1                                           COMATOK    1010    A
COMATOK    IFGT '?TKSTLK,999D,1                                             COMATOK    1011    A
COMATOK   P ERR TOO MANY *TLINK* ENTRIES "SEQUENCE"                         COMATOK    1012    A
COMATOK  *                                                                  COMATOK    1013    A
COMATOK   '?TKCS03 DECMIC '?TK3LEN,3                                        COMATOK    1014    A
COMATOK   '?TKCS03 MICRO 1,, '?TK3"'?TKCS03"                                COMATOK    1015    A
COMATOK  *                                                                  COMATOK    1016    A
COMATOK   '?TKCS04 DECMIC '?TKSTLK,3                                        COMATOK    1017    A
COMATOK   '?TKCTLK MICRO 1,, '?"'?TKCS01"                                   COMATOK    1018    A
COMATOK  *                                                                  COMATOK    1019    A
COMATOK   '?TK#003 ENDIF                                                    COMATOK    1020    A
COMATOK  *                                                                  COMATOK    1021    A
COMATOK  *         HERE WHEN READY TO MAKE ENTRY INTO '?TK3NNN.             COMATOK    1022    A
COMATOK  *                                                                  COMATOK    1023    A
COMATOK  *         ENTRY  '?TKCS01 = CHARS IN THIS TOKEN TYPE NAME.         COMATOK    1024    A
COMATOK  *                '?TKCS03 = '?TK3NNN ENTRY TO STORE INTO.          COMATOK    1025    A
COMATOK  *                '?TKCS04 = '?TKTNNN NR FOR THIS TOKEN.            COMATOK    1026    A
COMATOK  *                                                                  COMATOK    1027    A
COMATOK   "'?TKCS03" MICRO 1,, "'?TKCS04""'?TKCS01"                         COMATOK    1028    A
COMATOK  *                                                                  COMATOK    1029    A
COMATOK  *         READY TO POP *STX* ENTRIES OFF '?TK2NNN, GENERATING      COMATOK    1030    A
COMATOK  *         ENTRIES ALONG THE WAY.                                   COMATOK    1031    A
COMATOK  *                                                                  COMATOK    1032    A
COMATOK     IFLE '?TK2LEN,1,1                                               COMATOK    1033    A
COMATOK   P ERR *STX* PARAMETER MUST CONTAIN AT LEAST 2 TOKENS "SEQUENCE"   COMATOK    1034    A
COMATOK  *                                                                  COMATOK    1035    A
COMATOK   '?TKCFAL MICRO 1,, '?TKT"'?TKCS04"                                COMATOK    1036    A
COMATOK  *                                                                  COMATOK    1037    A
COMATOK   '?TK#006 DUP '?TK2LEN-1                                           COMATOK    1038    A
COMATOK  *                                                                  COMATOK    1039    A
COMATOK   '?TK2LEN SET '?TK2LEN-1                                           COMATOK    1040    A
COMATOK   '?TKCS01 DECMIC '?TK2LEN,3                                        COMATOK    1041    A
COMATOK   '?TKCTOT '?TKMMEQ '?TK2"'?TKCS01"                                 COMATOK    1042    A
COMATOK  *                                                                  COMATOK    1043    A
COMATOK   '?TKSTLK SET '?TKSTLK+1                                           COMATOK    1044    A
COMATOK    IFGT '?TKSTLK,999D,1                                             COMATOK    1045    A
COMATOK   P ERR TOO MANY *TLINK* ENTRIES "SEQUENCE"                         COMATOK    1046    A
COMATOK  *                                                                  COMATOK    1047    A
COMATOK   '?TKCTRU DECMIC '?TKSTLK,3                                        COMATOK    1048    A
COMATOK   "'?TKCTLK" '?TKMTLK "'?TKCTOT","?TKT"'?TKCTRU","'?TKCFAL"         COMATOK    1049    A
COMATOK  *                                                                  COMATOK    1050    A
COMATOK   '?TKCTLK MICRO 1,, '?TKT"'?TKCTRU"                                COMATOK    1051    A
COMATOK   '?TK#006 ENDD                                                     COMATOK    1052    A
COMATOK  *                                                                  COMATOK    1053    A
COMATOK   '?TKT"'?TKCTRU" SET 0                                             COMATOK    1054    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

```
COMATOK  *                                                                  COMATOK    1055    A
COMATOK   '?TKCS01 MICRO 1,, D                                              COMATOK    1056    A
COMATOK   '?TKCS02 MICRO 1,1, D                                             COMATOK    1057    A
COMATOK  *                                                                  COMATOK    1058    A
COMATOK    IFC EQ, "'?TKCS02" * ,2                                          COMATOK    1059    A
COMATOK   '?TKCS01 MICRO 2,, D                                              COMATOK    1060    A
COMATOK   '?TKCS01 MICRO 1,, -"'?TKCS01"                                    COMATOK    1061    A
COMATOK  *                                                                  COMATOK    1062    A
COMATOK   A '?TKMTOK BB,("'?TKCS01"),,'?TKRTLK                              COMATOK    1063    A
COMATOK  *                                                                  COMATOK    1064    A
COMATOK   '?TK#001 ENDIF                                                    COMATOK    1065    A
COMATOK  *                                                                  COMATOK    1066    A
COMATOK   TOKEN    ENDM                                                     COMATOK    1067    A
COMATOK   XEQ      SPACE  4,10                                              COMATOK    1068    A
COMATOK  **        *XEQ* TOGEL INSTRUCTION/MACRO.                           COMATOK    1069    A
COMATOK  *                                                                  COMATOK    1070    A
COMATOK  *         XEQ    PRN                                               COMATOK    1071    A
COMATOK  *                                                                  COMATOK    1072    A
COMATOK  *         ENTRY  PRN  = NAME/ADDR OF *TOGEL* PROC TO XEQ.          COMATOK    1073    A
COMATOK  *                                                                  COMATOK    1074    A
COMATOK  *         EXIT   NONE                                              COMATOK    1075    A
COMATOK  *                                                                  COMATOK    1076    A
COMATOK  *         USES   NONE                                              COMATOK    1077    A
COMATOK  *                                                                  COMATOK    1078    A
COMATOK  *         CALLS  NONE                                              COMATOK    1079    A
COMATOK                                                                     COMATOK    1080    A
COMATOK                                                                     COMATOK    1081    A
COMATOK            PURGMAC XEQ                                              COMATOK    1082    A
COMATOK                                                                     COMATOK    1083    A
COMATOK   XEQ      MACRO  P                                                 COMATOK    1084    A
COMATOK  *                                                                  COMATOK    1085    A
COMATOK  *         GENERATE *TOGEL* BINARY INSTRUCTION.                     COMATOK    1086    A
COMATOK  *                                                                  COMATOK    1087    A
COMATOK   '?TKS001 SET 60-TG.XADP-TG.XADL                                   COMATOK    1088    A
COMATOK   + VFD '?TKS001/0,TG.XADL/=X_P-FW.TOK,TG.TOFUL/=XTOK=XEQ-FW.TOK    COMATOK    1089    A
COMATOK  *                                                                  COMATOK    1090    A
COMATOK   XEQ      ENDM                                                     COMATOK    1091    A
COMATOK   '?TKMAPL SPACE  4,10                                              COMATOK    1092    A
COMATOK  **        '?TKMAPL - GENERATE APLIST FOR *CHMIC*.                  COMATOK    1093    A
COMATOK  *                                                                  COMATOK    1094    A
COMATOK  *                                                                  COMATOK    1095    A
COMATOK  *         ENTRY  A = BINARY DPC VALUE OF CHAR TO GEN APL FOR       COMATOK    1096    A
COMATOK  *                                                                  COMATOK    1097    A
COMATOK  *         EXIT   '?TKCAPL UPDATED                                  COMATOK    1098    A
COMATOK  *                                                                  COMATOK    1099    A
COMATOK  *         CALLS  NONE                                              COMATOK    1100    A
COMATOK                                                                     COMATOK    1101    A
COMATOK                                                                     COMATOK    1102    A
COMATOK   '?TKMAPL MACRO A                                                  COMATOK    1103    A
COMATOK   '?TKCS01 OCTMIC A,2                                               COMATOK    1104    A
COMATOK   '?TKCAPL MICRO 1,, "'?TKCAPL","'?TKCS01"                          COMATOK    1105    A
COMATOK   '?TKMAPL ENDM                                                     COMATOK    1106    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

```
COMATOK   '?TKMGRT SPACE  4,10                                          COMATOK     1107     A
COMATOK  **        '?TKMGRT - GET RANGE TOKEN.                          COMATOK     1108     A
COMATOK  *                                                              COMATOK     1109     A
COMATOK  *                                                              COMATOK     1110     A
COMATOK  *         THIS MACRO PROVIDES *'?TKMRNG* WITH A SINGLE RANGE TOKEN.  COMATOK  1111  A
COMATOK  *                                                              COMATOK     1112     A
COMATOK  *         ENTRY  '?TKCICS = CHAR STRING                        COMATOK     1113     A
COMATOK  *                '?TKPICS = ORD OF 1ST CHAR IN "'?TKCICS" TO GENERATE  COMATOK  1114  A
COMATOK  *                           TOKEN FOR.                         COMATOK     1115     A
COMATOK  *                                                              COMATOK     1116     A
COMATOK  *         EXIT   '?TKCICS = UNCHANGED                          COMATOK     1117     A
COMATOK  *                '?TKSTOT = TOKEN TYPE.  TOKEN TYPES ARE --    COMATOK     1118     A
COMATOK  *                         = 0 IF END-OF-STRING ENCOUNTERED.    COMATOK     1119     A
COMATOK  *                         = 1 IF SINGLE CHAR TOKEN.  SEE *'?TKSDPC*  COMATOK  1120  A
COMATOK  *                             BELOW.                           COMATOK     1121     A
COMATOK  *                         = 2 IF *..* TOKEN ENCOUNTERED.       COMATOK     1122     A
COMATOK  *                '?TKSDPC = BINARY DPC VALUE OF CHAR IF *'?TKSTOT=1*.  COMATOK  1123  A
COMATOK  *                                                              COMATOK     1124     A
COMATOK  *         CALLS  NONE                                          COMATOK     1125     A
COMATOK                                                                 COMATOK     1126     A
COMATOK                                                                 COMATOK     1127     A
COMATOK   '?TKMGRT MACRO                                                COMATOK     1128     A
COMATOK  *                                                              COMATOK     1129     A
COMATOK  *         CHECK FOR END-OF-STRING TOKEN.                       COMATOK     1130     A
COMATOK  *                                                              COMATOK     1131     A
COMATOK    IFGT '?TKPICS,'?TKLICS,2                                     COMATOK     1132     A
COMATOK   '?TKSTOT SET 0                                                COMATOK     1133     A
COMATOK   '?TK#002 SKIP                                                 COMATOK     1134     A
COMATOK  *                                                              COMATOK     1135     A
COMATOK  *         CHECK FOR *..* TOKEN.                                COMATOK     1136     A
COMATOK  *                                                              COMATOK     1137     A
COMATOK    IFEQ '?TKSTOT,2,1                                            COMATOK     1138     A
COMATOK   '?TK#001 SKIP                                                 COMATOK     1139     A
COMATOK    IFLE '?TKPICS,'?TKLICS-1,5                                   COMATOK     1140     A
COMATOK   '?TKCS01 MICRO '?TKPICS,2, "'?TKCICS"                         COMATOK     1141     A
COMATOK    IFC EQ, "'?TKCS01" .. ,3                                     COMATOK     1142     A
COMATOK   '?TKPICS SET '?TKPICS+2                                       COMATOK     1143     A
COMATOK   '?TKSTOT SET 2                                                COMATOK     1144     A
COMATOK   '?TK#002 SKIP                                                 COMATOK     1145     A
COMATOK  *                                                              COMATOK     1146     A
COMATOK  *         CHECK FOR */NN* TOKEN.                               COMATOK     1147     A
COMATOK  *                                                              COMATOK     1148     A
COMATOK    IFLE '?TKPICS,'?TKLICS-2,12D                                 COMATOK     1149     A
COMATOK   '?TKCS01 MICRO '?TKPICS,1, "'?TKCICS"                         COMATOK     1150     A
COMATOK    IFC NE, "'?TKCS01" / ,1                                      COMATOK     1151     A
COMATOK   '?TK#001 SKIP                                                 COMATOK     1152     A
COMATOK   '?TKCS01 MICRO '?TKPICS+1,2, "'?TKCICS"                       COMATOK     1153     A
COMATOK    IFC GT, "'?TKCS01" 77 ,1                                     COMATOK     1154     A
COMATOK   '?TK#001 SKIP                                                 COMATOK     1155     A
COMATOK    IFC LT, "'?TKCS01" 00 ,1                                     COMATOK     1156     A
COMATOK   '?TK#001 SKIP                                                 COMATOK     1157     A
COMATOK   '?TKSTOT SET 1                                                COMATOK     1158     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

```
COMATOK   '?TKSDPC SET "'?TKCS01"B                                  COMATOK    1159    A
COMATOK   '?TKPICS SET '?TKPICS+3                                   COMATOK    1160    A
COMATOK   '?TK#002 SKIP                                             COMATOK    1161    A
COMATOK  *                                                          COMATOK    1162    A
COMATOK  *         HERE IF THIS IS A SINGLE CHAR TOKEN.             COMATOK    1163    A
COMATOK  *                                                          COMATOK    1164    A
COMATOK   '?TK#001 ENDIF                                            COMATOK    1165    A
COMATOK   '?TKCS01 MICRO '?TKPICS,1, "'?TKCICS"                     COMATOK    1166    A
COMATOK   '?TKSTOT SET 1                                            COMATOK    1167    A
COMATOK   '?TKSDPC SET 1R"'?TKCS01"                                 COMATOK    1168    A
COMATOK   '?TKPICS SET '?TKPICS+1                                   COMATOK    1169    A
COMATOK  *                                                          COMATOK    1170    A
COMATOK  *         HERE WHEN DONE.                                  COMATOK    1171    A
COMATOK  *                                                          COMATOK    1172    A
COMATOK   '?TK#002 ENDIF                                            COMATOK    1173    A
COMATOK   '?TKMGRT ENDM                                             COMATOK    1174    A
COMATOK   '?TKMMEQ SPACE  4,10                                      COMATOK    1175    A
COMATOK  **        '?TKMMEQ - MICRO EQUIVALENCE.                    COMATOK    1176    A
COMATOK  *                                                          COMATOK    1177    A
COMATOK  *                                                          COMATOK    1178    A
COMATOK  *         THIS MACRO EQUATES TWO MICRO STRINGS.  IT IS ONLY TRULY  COMATOK    1179    A
COMATOK  *         USEFUL WHEN THE MICRO NAME ON THE RIGHT-SIDE OF THE  COMATOK    1180    A
COMATOK  *         EQUIVALENCE IS BEING GENERATED VIA MICRO STRINGS.  COMATOK    1181    A
COMATOK  *                                                          COMATOK    1182    A
COMATOK  * A       '?TKMMEQ B                                       COMATOK    1183    A
COMATOK  *                                                          COMATOK    1184    A
COMATOK  *         ENTRY  A = MICRO NAME TO EQUATE *B* TO.          COMATOK    1185    A
COMATOK  *                B = OLD MICRO NAME.                       COMATOK    1186    A
COMATOK  *                                                          COMATOK    1187    A
COMATOK  *         EXIT   "A" CREATED AND "B" LEFT ALONE.           COMATOK    1188    A
COMATOK  *                                                          COMATOK    1189    A
COMATOK  *         CALLS  NONE                                      COMATOK    1190    A
COMATOK                                                             COMATOK    1191    A
COMATOK                                                             COMATOK    1192    A
COMATOK            PURGMAC '?TKMMEQ                                 COMATOK    1193    A
COMATOK                                                             COMATOK    1194    A
COMATOK    MACRO '?TKMMEQ,A,B                                       COMATOK    1195    A
COMATOK   A MICRO 1,, "B"                                           COMATOK    1196    A
COMATOK   '?TKMMEQ ENDM                                             COMATOK    1197    A
COMATOK   '?TKMPMA SPACE  4,10                                      COMATOK    1198    A
COMATOK  **        '?TKMPMA - PRINT MICRO ARRAY.                    COMATOK    1199    A
COMATOK  *                                                          COMATOK    1200    A
COMATOK  *                                                          COMATOK    1201    A
COMATOK  *         THIS MACRO IS FOR INTERNAL DEBUGGING ONLY.  WHEN USED,  COMATOK    1202    A
COMATOK  *         PRINTS THE CONTENTS OF A SPECIFIED MICRO ARRAY.  A MICRO  COMATOK    1203    A
COMATOK  *         ARRAY HAS THE FORM --                            COMATOK    1204    A
COMATOK  *                                                          COMATOK    1205    A
COMATOK  *                '?TKNXXX                                  COMATOK    1206    A
COMATOK  *                                                          COMATOK    1207    A
COMATOK  *         WHERE *'?TK* IS A *COMATOK* COMMON SYMBOL PREFIX, *N* IS  COMATOK    1208    A
COMATOK  *         A MICRO ARRAY NUMBER, AND *XXX* IS A MICRO ARRAY ELEMENT  COMATOK    1209    A
COMATOK  *         NUMBER.  FOR EXAMPLE, CONSIDER THE MICRO ARRAY NUMBER *3*.  COMATOK    1210    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

```
COMATOK  *         EACH ELEMENT IN MICRO ARRAY *3* IS A MICRO OF THE FORM --   COMATOK  1211  A
COMATOK  *                                                                     COMATOK  1212  A
COMATOK  *                '?TK3XXX                                             COMATOK  1213  A
COMATOK  *                                                                     COMATOK  1214  A
COMATOK  *         I.E. -- '?TK3001, '?TK3002, '?TK3003, ETC.                  COMATOK  1215  A
COMATOK  *                                                                     COMATOK  1216  A
COMATOK  *         '?TKMPMA N                                                  COMATOK  1217  A
COMATOK  *                                                                     COMATOK  1218  A
COMATOK  *         ENTRY  N = MICRO ARRAY NR.                                  COMATOK  1219  A
COMATOK  *                                                                     COMATOK  1220  A
COMATOK  *         EXIT   NONE                                                 COMATOK  1221  A
COMATOK  *                                                                     COMATOK  1222  A
COMATOK  *         USES   NONE                                                 COMATOK  1223  A
COMATOK  *                                                                     COMATOK  1224  A
COMATOK  *         CALLS  '?TKMMEQ                                             COMATOK  1225  A
COMATOK                                                                        COMATOK  1226  A
COMATOK                                                                        COMATOK  1227  A
COMATOK            PURGMAC '?TKMPMA                                            COMATOK  1228  A
COMATOK                                                                        COMATOK  1229  A
COMATOK   '?TKMPMA MACRO N                                                     COMATOK  1230  A
COMATOK    LOCAL '?TK#001,'?TK#002                                             COMATOK  1231  A
COMATOK  *                                                                     COMATOK  1232  A
COMATOK   '?TKSD01 SET 0                                                       COMATOK  1233  A
COMATOK  *                                                                     COMATOK  1234  A
COMATOK   '?TK#002 DUP '?TK_N_LEN                                              COMATOK  1235  A
COMATOK   '?TKSD01 SET '?TKSD01+1                                              COMATOK  1236  A
COMATOK  *                                                                     COMATOK  1237  A
COMATOK   '?TKCD01 DECMIC '?TKSD01,3                                           COMATOK  1238  A
COMATOK   '?TKCD02 '?TKMMEQ '?TK_N_"'?TKCD01"                                  COMATOK  1239  A
COMATOK  *                                                                     COMATOK  1240  A
COMATOK    '?TK_N_"'?TKCD01" = "'?TKCD02"                                      COMATOK  1241  A
COMATOK  *                                                                     COMATOK  1242  A
COMATOK   '?TK#002 ENDD                                                        COMATOK  1243  A
COMATOK  *                                                                     COMATOK  1244  A
COMATOK   '?TKMPMA ENDM                                                        COMATOK  1245  A
COMATOK   '?TKMRMT SPACE  4,10                                                 COMATOK  1246  A
COMATOK  **        '?TKMRMT - FORCE MICRO EVALUATION FOR *RMT* BLOCKS.         COMATOK  1247  A
COMATOK  *                                                                     COMATOK  1248  A
COMATOK  *                                                                     COMATOK  1249  A
COMATOK  *         THIS MACRO IS USED SOLEY FOR THE PURPOSE OF FORCING         COMATOK  1250  A
COMATOK  *         MICRO EVALUATION IN *RMT* BLOCKS.                           COMATOK  1251  A
COMATOK  *                                                                     COMATOK  1252  A
COMATOK  * BLK     '?TKMRMT S                                                  COMATOK  1253  A
COMATOK  *                                                                     COMATOK  1254  A
COMATOK  *         ENTRY  BLK = REMOTE BLOCK NAME                              COMATOK  1255  A
COMATOK  *                S   = STMT TO BE ENTERED INTO REMOTE BLOCK *BLK*.    COMATOK  1256  A
COMATOK  *                                                                     COMATOK  1257  A
COMATOK  *         EXIT   NONE                                                 COMATOK  1258  A
COMATOK  *                                                                     COMATOK  1259  A
COMATOK  *         USES   NONE                                                 COMATOK  1260  A
COMATOK  *                                                                     COMATOK  1261  A
COMATOK  *         CALLS  NONE                                                 COMATOK  1262  A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

```
COMATOK                                                                  COMATOK    1263    A
COMATOK                                                                  COMATOK    1264    A
COMATOK            PURGMAC '?TKMRMT                                      COMATOK    1265    A
COMATOK                                                                  COMATOK    1266    A
COMATOK            MACRO  '?TKMRMT,BLK,S                                 COMATOK    1267    A
COMATOK   BLK RMT                                                        COMATOK    1268    A
COMATOK   S                                                              COMATOK    1269    A
COMATOK   BLK RMT                                                        COMATOK    1270    A
COMATOK   '?TKMRMT ENDM                                                  COMATOK    1271    A
COMATOK   '?TKMRNG SPACE  4,10                                           COMATOK    1272    A
COMATOK  **        '?TKMRNG - GENERATE MICRO FOR CHARACTER RANGE.        COMATOK    1273    A
COMATOK  *                                                               COMATOK    1274    A
COMATOK  *                                                               COMATOK    1275    A
COMATOK  *         THIS MACRO EVALUATES THE *TOGEL* CHARACTER RANGE, AS IT COMATOK    1276    A
COMATOK  *         OCCURS IN *GROUP* AND *IFT* STATEMENTS, AND CREATES THE COMATOK    1277    A
COMATOK  *         MICRO "'?TKCRNG" WHICH CONSISTS OF A BIT STRING THAT HAS COMATOK    1278    A
COMATOK  *         A BIT *ON* FOR EACH CHARACTER IN THE RANGE.  THIS BIT COMATOK    1279    A
COMATOK  *         STRING IN "'?TKCRNG" IS DESTINED TO BECOME A          COMATOK    1280    A
COMATOK  *         CHARACTER SHIFT MASK AS IS USED BY *COMCBUB*.         COMATOK    1281    A
COMATOK  *         SEE *COMCTOK* AND *COMCBUB/COMCBUN*.  NOTE THAT CHARS COMATOK    1282    A
COMATOK  *         IN "'?TKCRNG" ARE THEREFORE IN A *ROTATED CHAR SET*   COMATOK    1283    A
COMATOK  *         REPRESENTATION.                                       COMATOK    1284    A
COMATOK  *                                                               COMATOK    1285    A
COMATOK  *         THE ALGORITHM USED HERE IS TO TREAT THE CHARACTERS    COMATOK    1286    A
COMATOK  *         IN THE RANGE (I.E. THE INPUT TO THIS MACRO) AS A STRING COMATOK    1287    A
COMATOK  *         THAT IS TO BE SCANNED FROM LEFT-TO-RIGHT (VIA MICRO   COMATOK    1288    A
COMATOK  *         MANIPULATION).  THIS IS DONE BY MAKING CONSECUTIVE    COMATOK    1289    A
COMATOK  *         CALLS TO *'?TKMGRT* (GET RANGE TOKEN) WHICH RETURNS A COMATOK    1290    A
COMATOK  *         TOKEN TYPE, *'?TKSTOT*, FOR THE NEXT TOKEN IN THE RANGE. COMATOK    1291    A
COMATOK  *                                                               COMATOK    1292    A
COMATOK  *         TOKEN TYPES, *'?TKSTOT*, ARE--                        COMATOK    1293    A
COMATOK  *                                                               COMATOK    1294    A
COMATOK  *               = 0 IF END-OF-STRING ENCOUNTERED.  I.E. WE ARE DONE. COMATOK    1295    A
COMATOK  *               = 1 IF CHAR TOKEN ENCOUNTERED.  THE ACTUAL CHAR IS IN COMATOK    1296    A
COMATOK  *                   *'?TKSDPC* (TOKEN VALUE) IN BINARY DPC.  *'?TKMGRT* COMATOK    1297    A
COMATOK  *                   WILL CALL *'?TKMGRB* (GENERATE RANGE BIT) TO MERGE COMATOK    1298    A
COMATOK  *                   THE APPROPRIATE BIT FOR THIS CHAR INTO "'?TKCRNG". COMATOK    1299    A
COMATOK  *               = 2 IF *..* TOKEN ENCOUNTERED.  *'?TKMGRT* WILL CALL COMATOK    1300    A
COMATOK  *                   '?TKMAPL TO GENERATE APLIST ITEMS FOR     COMATOK    1301    A
COMATOK  *                   EACH CHAR IN RANGE, TO BE USED AS INPUT   COMATOK    1302    A
COMATOK  *                   TO *CHMIC*.                               COMATOK    1303    A
COMATOK  *                                                               COMATOK    1304    A
COMATOK  *         IT SHOULD BE NOTED THAT THE */NN* SYNTAX IS RESOLVED BY COMATOK    1305    A
COMATOK  *         *'?TKMGRT*, WHICH RETURNS *'?TKSTOT=1* WITH THE BINARY COMATOK    1306    A
COMATOK  *         VALUE *NN* IN *'?TKSDPC*.                             COMATOK    1307    A
COMATOK  *                                                               COMATOK    1308    A
COMATOK  *         ENTRY  A = CHARACTER RANGE AS IT OCCURS IN *TOGEL*.   COMATOK    1309    A
COMATOK  *                                                               COMATOK    1310    A
COMATOK  *         EXIT   '?TKCRNG = CHAR SHIFT MASK FOR *A*.            COMATOK    1311    A
COMATOK  *                                                               COMATOK    1312    A
COMATOK  *         CALLS  '?TKMAPL,'?TKMGRT,CHMIC                        COMATOK    1313    A
COMATOK                                                                  COMATOK    1314    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

```
COMATOK                                                                  COMATOK     1315     A
COMATOK   '?TKMRNG MACRO A                                               COMATOK     1316     A
COMATOK  *                                                               COMATOK     1317     A
COMATOK  *         INITIALIZE --                                         COMATOK     1318     A
COMATOK  *                                                               COMATOK     1319     A
COMATOK  *         '?TKCICS = MICRO THAT CONTAINS THE ENTIRE INPUT CHARACTER  COMATOK  1320     A
COMATOK  *                    STRING.                                    COMATOK     1321     A
COMATOK  *         '?TKPICS = ORD OF NEXT CHAR IN "'?TKCICS".            COMATOK     1322     A
COMATOK  *         '?TKLICS = LEN OF "'?TKCICS"                          COMATOK     1323     A
COMATOK  *         '?TKCSGN = MICRO THAT CONTAINS /-/ IF 1ST CHAR IN RANGE  COMATOK  1324     A
COMATOK  *                    IS /-/, I.E. PROGRAMMER SPECIFIED *(-XXX)* WHICH  COMATOK  1325     A
COMATOK  *                    MEANS THAT RANGE IS TO BE NEGATED.  ELSE NULL.  COMATOK  1326     A
COMATOK  *         '?TKCAPL = NULL.                                      COMATOK     1327     A
COMATOK  *         '?TKODPC = BIN DPC VALUE FOR ":"                      COMATOK     1328     A
COMATOK  *         '?TKSDPC = BIN DPC VALUE FOR ":"                      COMATOK     1329     A
COMATOK  *         '?TKSTOT = 0, I.E. BEGINNING-OF-STRING.               COMATOK     1330     A
COMATOK  *                                                               COMATOK     1331     A
COMATOK   '?TKCAPL MICRO                                                 COMATOK     1332     A
COMATOK   '?TKCICS MICRO 1,, A                                           COMATOK     1333     A
COMATOK   '?TKCSGN MICRO 1,1, "'?TKCICS"                                 COMATOK     1334     A
COMATOK  *                                                               COMATOK     1335     A
COMATOK   '?TK#001 IFC NE, "'?TKCSGN" -                                  COMATOK     1336     A
COMATOK   '?TKCSGN MICRO                                                 COMATOK     1337     A
COMATOK   '?TK#001 ELSE                                                  COMATOK     1338     A
COMATOK   '?TKCICS MICRO 2,, "'?TKCICS"                                  COMATOK     1339     A
COMATOK   '?TK#001 ENDIF                                                 COMATOK     1340     A
COMATOK  *                                                               COMATOK     1341     A
COMATOK   '?TKPICS SET 1                                                 COMATOK     1342     A
COMATOK   '?TKLICS MICCNT '?TKCICS                                       COMATOK     1343     A
COMATOK   '?TKODPC SET 1R:                                               COMATOK     1344          I
         -CCGA015
COMATOK   '?TKSDPC SET 1R:                                               COMATOK     1345          I
         -CCGA015
COMATOK   '?TKODPC SET    0                                              CCGA015        4     A
COMATOK   '?TKSDPC SET    0                                              CCGA015        5     A
COMATOK  *                                                               COMATOK     1346     A
COMATOK   '?TKSTOT SET 0                                                 COMATOK     1347     A
COMATOK  *                                                               COMATOK     1348     A
COMATOK  *         CYCLE THRU CHARACTER RANGE IN "'?TKCICS",             COMATOK     1349     A
COMATOK  *         GENERATING "'?TKCAPL".                                COMATOK     1350     A
COMATOK  *                                                               COMATOK     1351     A
COMATOK   '?TK#006 DUP -1                                                COMATOK     1352     A
COMATOK    '?TKMGRT                                                      COMATOK     1353     A
COMATOK  *                                                               COMATOK     1354     A
COMATOK  *         IF NOT END-OF-STRING, THEN PROCESS NEXT TOKEN.        COMATOK     1355     A
COMATOK  *         ELSE READY TO GENERATE "'?TKCRNG" USING "'?TKCAPL".   COMATOK     1356     A
COMATOK  *                                                               COMATOK     1357     A
COMATOK   '?TK#005 IFNE '?TKSTOT,0                                       COMATOK     1358     A
COMATOK  *                                                               COMATOK     1359     A
COMATOK  *         CHECK FOR AND PROCESS *..* TOKEN.                     COMATOK     1360     A
COMATOK  *                                                               COMATOK     1361     A
COMATOK   '?TK#003 IFEQ '?TKSTOT,2                                       COMATOK     1362     A

         0        1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

```
COMATOK  *                                                          COMATOK   1363   A
COMATOK  *         HERE IF *..* TOKEN OCCURRED.                     COMATOK   1364   A
COMATOK  *         GET TOKEN/CHAR THAT TERMINATES RANGE.            COMATOK   1365   A
COMATOK  *                                                          COMATOK   1366   A
COMATOK    '?TKMGRT                                                 COMATOK   1367   A
COMATOK    IFEQ '?TKSTOT,0,1                                        COMATOK   1368   A
COMATOK   '?TKSDPC SET 1R;                                          COMATOK   1369   A
COMATOK  *                                                          COMATOK   1370   A
COMATOK  *         LOOP THROUGH CHAR RANGE, GENERATING CHARS TO *'?TKCAPL*  COMATOK   1371   A
COMATOK  *         AS WE GO.  CHAR RANGE IS FROM *'?TKODPC* TO *'?TKSDPC*.  COMATOK   1372   A
COMATOK  *                                                          COMATOK   1373   A
COMATOK    '?TKMAPL '?TKODPC                                        COMATOK   1374   A
COMATOK   '?TK#001 DUP -1                                           COMATOK   1375   A
COMATOK   '?TK#002 IFNE '?TKODPC,'?TKSDPC                           COMATOK   1376   A
COMATOK   '?TKCS01 OCTMIC '?TKODPC+1+10000B,2                       COMATOK   1377   A
COMATOK   '?TKODPC SET "'?TKCS01"B                                  COMATOK   1378   A
COMATOK    '?TKMAPL '?TKODPC                                        COMATOK   1379   A
COMATOK   '?TK#002 ELSE                                             COMATOK   1380   A
COMATOK    STOPDUP                                                  COMATOK   1381   A
COMATOK   '?TK#002 ENDIF                                            COMATOK   1382   A
COMATOK   '?TK#001 ENDD                                             COMATOK   1383   A
COMATOK  *                                                          COMATOK   1384   A
COMATOK  *         CHECK FOR AND PROCESS A SINGLE CHAR TOKEN.       COMATOK   1385   A
COMATOK  *                                                          COMATOK   1386   A
COMATOK   '?TK#003 ELSE                                             COMATOK   1387   A
COMATOK     IFEQ '?TKSTOT,1,1                                       COMATOK   1388   A
COMATOK     '?TKMAPL '?TKSDPC                                       COMATOK   1389   A
COMATOK  *                                                          COMATOK   1390   A
COMATOK   '?TK#003 ENDIF                                            COMATOK   1391   A
COMATOK   '?TK#005 ELSE                                             COMATOK   1392   A
COMATOK     STOPDUP                                                 COMATOK   1393   A
COMATOK   '?TK#005 ENDIF                                            COMATOK   1394   A
COMATOK  *                                                          COMATOK   1395   A
COMATOK  *         NEW CHAR BECOMES OLD CHAR.                       COMATOK   1396   A
COMATOK  *                                                          COMATOK   1397   A
COMATOK   '?TKODPC SET '?TKSDPC                                     COMATOK   1398   A
COMATOK   '?TK#006 ENDD                                             COMATOK   1399   A
COMATOK  *                                                          COMATOK   1400   A
COMATOK  *         GENERATE "'?TKCRNG" MICRO.                       COMATOK   1401   A
COMATOK  *                                                          COMATOK   1402   A
COMATOK   '?TKCRNG CHMIC ("'?TKCAPL"),BICH                          COMATOK   1403   A
COMATOK   '?TKCRNG MICRO 1,, "'?TKCSGN""'?TKCRNG"                   COMATOK   1404   A
COMATOK   '?TKMRNG ENDM                                             COMATOK   1405   A
COMATOK   '?TKMSMA SPACE  4,10                                      COMATOK   1406   A
COMATOK  **        '?TKMSMA - SCAN MICRO ARRAY.                     COMATOK   1407   A
COMATOK  *                                                          COMATOK   1408   A
COMATOK  *                                                          COMATOK   1409   A
COMATOK  *         '?TKMSMA N,FC,NC                                 COMATOK   1410   A
COMATOK  *                                                          COMATOK   1411   A
COMATOK  *         ENTRY  N  - MICRO ARRAY NR.                      COMATOK   1412   A
COMATOK  *                FC = 1ST CHAR POSITION IN EACH ARRAY ELEMENT  COMATOK   1413   A
COMATOK  *                     TO SEARCH AT.                        COMATOK   1414   A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

```
COMATOK  *                NC = NR OF CHARS (BEGINNING AT *FC*) TO COMPARE,          COMATOK   1415   A
COMATOK  *                   = 0 IF REST OF MICRO ARRAY ENTRY (AS IN *MICRO*        COMATOK   1416   A
COMATOK  *                     PSEUDO).                                             COMATOK   1417   A
COMATOK  *                                                                          COMATOK   1418   A
COMATOK  *                '?TKCS01 = ELEMENT TO SEARCH FOR.                         COMATOK   1419   A
COMATOK  *                '?TKNLEN = NR OF ENTRIES IN THIS MICRO ARRAY              COMATOK   1420   A
COMATOK  *                           (WHERE *N* IS MICRO ARRAY NR).                 COMATOK   1421   A
COMATOK  *                                                                          COMATOK   1422   A
COMATOK  *         EXIT   '?TKCS01 = UNCHANGED.                                     COMATOK   1423   A
COMATOK  *                '?TKS001 = MICRO ARRAY ELEMENT NR IF A FIND, ELSE         COMATOK   1424   A
COMATOK  *                         = .ZR. IF NO FIND.                               COMATOK   1425   A
COMATOK  *                '?TKCS02 = MICRO CONTAINING FOUND ENTRY IN '?TKNXXX.      COMATOK   1426   A
COMATOK  *                '?TKCS03 = MICRO CONTAINING MICRO NAME OF FOUND           COMATOK   1427   A
COMATOK  *                           ENTRY.                                         COMATOK   1428   A
COMATOK  *                           I.E. "'?TKCS03" MICRO 1,, "'?TKCS02"           COMATOK   1429   A
COMATOK  *                                                                          COMATOK   1430   A
COMATOK  *         CALLS  '?TKMMEQ                                                  COMATOK   1431   A
COMATOK                                                                             COMATOK   1432   A
COMATOK                                                                             COMATOK   1433   A
COMATOK            PURGMAC '?TKMSMA                                                 COMATOK   1434   A
COMATOK                                                                             COMATOK   1435   A
COMATOK   '?TKMSMA MACRO A,B,C                                                      COMATOK   1436   A
COMATOK    LOCAL '?TK#001,'?TK#002,'?TK#003                                         COMATOK   1437   A
COMATOK  *                                                                          COMATOK   1438   A
COMATOK   '?TKS001 SET '?TK_A_LEN                                                   COMATOK   1439   A
COMATOK   '?TK#002 IFGT '?TKS001,0                                                  COMATOK   1440   A
COMATOK  *                                                                          COMATOK   1441   A
COMATOK   '?TK#001 DUP '?TK_A_LEN                                                   COMATOK   1442   A
COMATOK  *                                                                          COMATOK   1443   A
COMATOK   '?TKCS03 DECMIC '?TKS001,3                                                COMATOK   1444   A
COMATOK   '?TKCS03 MICRO 1,, '?TK_A_"'?TKCS03"                                      COMATOK   1445   A
COMATOK   '?TKCS02 '?TKMMEQ "'?TKCS03"                                              COMATOK   1446   A
COMATOK   '?TKCS04 MICRO B,C, "'?TKCS02"                                            COMATOK   1447   A
COMATOK   '?TK#003  IFC EQ, "'?TKCS01" "'?TKCS04"                                   COMATOK   1448   A
COMATOK    STOPDUP                                                                  COMATOK   1449   A
COMATOK  *                                                                          COMATOK   1450   A
COMATOK   '?TK#003 ELSE                                                             COMATOK   1451   A
COMATOK   '?TKS001 SET '?TKS001-1                                                   COMATOK   1452   A
COMATOK   '?TK#003 ENDIF                                                            COMATOK   1453   A
COMATOK   '?TK#001 ENDD                                                             COMATOK   1454   A
COMATOK   '?TK#002 ENDIF                                                            COMATOK   1455   A
COMATOK  *                                                                          COMATOK   1456   A
COMATOK   '?TKMSMA ENDM                                                             COMATOK   1457   A
COMATOK   '?TKMTLK SPACE  4,10                                                      COMATOK   1458   A
COMATOK  **        '?TKMTLK - GENERATE *TLINK* ENTRY.                               COMATOK   1459   A
COMATOK  *                                                                          COMATOK   1460   A
COMATOK  *                                                                          COMATOK   1461   A
COMATOK  *         THIS MACRO IS USED SOLELY TO FORCE MICRO EVALUATION              COMATOK   1462   A
COMATOK  *         OF PARAMETERS INTO THIS MACRO.                                   COMATOK   1463   A
COMATOK  *                                                                          COMATOK   1464   A
COMATOK  * LAB     '?TKMTLK TOT,TRU,FAL                                             COMATOK   1465   A
COMATOK  *                                                                          COMATOK   1466   A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

```
COMATOK  *         ENTRY  LAB = *TLINK* LABEL FOR THIS ENTRY.                COMATOK    1467    A
COMATOK  *                TOT = *STX* TOKEN FROM *TOKEN* MACRO.              COMATOK    1468    A
COMATOK  *                TRU = LABEL FOR *TRUE* LINK.                       COMATOK    1469    A
COMATOK  *                FAL = LABEL FOR *FALSE* LINK.                      COMATOK    1470    A
COMATOK  *                                                                   COMATOK    1471    A
COMATOK  *         EXIT   NONE                                               COMATOK    1472    A
COMATOK  *                                                                   COMATOK    1473    A
COMATOK  *         USES   NONE                                               COMATOK    1474    A
COMATOK  *                                                                   COMATOK    1475    A
COMATOK  *         CALLS  NONE                                               COMATOK    1476    A
COMATOK                                                                      COMATOK    1477    A
COMATOK            PURGMAC '?TKMTLK                                          COMATOK    1478    A
COMATOK                                                                      COMATOK    1479    A
COMATOK            MACRO  '?TKMTLK,A,B,C,D                                   COMATOK    1480    A
COMATOK  *                                                                   COMATOK    1481    A
COMATOK   '?TKRTLK RMT                                                       COMATOK    1482    A
COMATOK   '?TKCS01 MICRO 1,, B                                               COMATOK    1483    A
COMATOK   '?TKS001 MICCNT '?TKCS01                                           COMATOK    1484    A
COMATOK  *                                                                   COMATOK    1485    A
COMATOK   '?TKCS02 MICRO 1,,'_B_'                                            COMATOK    1486    A
COMATOK   '?TKS002 MICCNT '?TKCS02                                           COMATOK    1487    A
COMATOK  *                                                                   COMATOK    1488    A
COMATOK   '?TKSMAF SET 0                                                     COMATOK    1489    A
COMATOK  *                                                                   COMATOK    1490    A
COMATOK   '?TK#007 IFNE '?TKS001,'?TKS002                                    COMATOK    1491    A
COMATOK  *                                                                   COMATOK    1492    A
COMATOK  *         HERE IF NEED TO PROCESS SPECIAL CHARACTER MATCH FORM.     COMATOK    1493    A
COMATOK  *                                                                   COMATOK    1494    A
COMATOK   '?TKSMAF SET 1                                                     COMATOK    1495    A
COMATOK  *                                                                   COMATOK    1496    A
COMATOK   '?TKCS03 MICRO '?TKS002+1,, B                                      COMATOK    1497    A
COMATOK   '?TKCS03 MICRO 1,,"'?TKCS03"'                                      COMATOK    1498    A
COMATOK  *                                                                   COMATOK    1499    A
COMATOK   '?TK#007 ENDIF                                                     COMATOK    1500    A
COMATOK  *                                                                   COMATOK    1501    A
COMATOK  *         GENERATE *TLINK* ENTRIES.                                 COMATOK    1502    A
COMATOK  *                                                                   COMATOK    1503    A
COMATOK    BSS 0                                                             COMATOK    1504          I
        -CCGA022
COMATOK   A SET *                                                            COMATOK    1505          I
        -CCGA022
COMATOK   A BSS 0                                                            CCGA022       5    A
COMATOK    VFD TL.MAFL/'?TKSMAF,5/0,TL.TRUL/C-FW.TOK,TL.FALL/D-FW.TOK,___________  COMATOK    1506    A
COMATOK  ,__TL.TOTL/"O."'?TKCS02"                                            COMATOK    1507    A
COMATOK  *                                                                   COMATOK    1508    A
COMATOK    IFEQ '?TKSMAF,1,1                                                 COMATOK    1509    A
COMATOK    DATA 0L"'?TKCS03"                                                 COMATOK    1510    A
COMATOK  *                                                                   COMATOK    1511    A
COMATOK   '?TKRTLK RMT                                                       COMATOK    1512    A
COMATOK  *                                                                   COMATOK    1513    A
COMATOK   '?TKMTLK ENDM                                                      COMATOK    1514    A
COMATOK   '?TKMTOK SPACE  4,10                                               COMATOK    1515    A

        0         1         2         3         4         5         6         7         8
        12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMATOK

COMATOK  **        '?TKMTOK - GENERATE *TOKEN* TABLE.                    COMATOK     1516     A
COMATOK  *                                                               COMATOK     1517     A
COMATOK  *                                                               COMATOK     1518     A
COMATOK  *         THIS MACRO IS USED SOLELY TO FORCE MICRO EVALUATION   COMATOK     1519     A
COMATOK  *         OF PARAMETERS INTO THIS MACRO.                        COMATOK     1520     A
COMATOK  *                                                               COMATOK     1521     A
COMATOK  * LAB     '?TKMTOK TOT,PAD,TAD,RMT                              COMATOK     1522     A
COMATOK  *                                                               COMATOK     1523     A
COMATOK  *         ENTRY  LAB = PROGRAMMER SPECIFIED LABEL FOR THIS *TOKEN*.  COMATOK  1524  A
COMATOK  *                TOT = TOKEN TYPE.                              COMATOK     1525     A
COMATOK  *                PAD = PROCESSOR ADDR.                          COMATOK     1526     A
COMATOK  *                TAD = TOKEN ADDR.  USED SO THAT *CASEOF* MACRO CAN  COMATOK  1527  A
COMATOK  *                      POINT TO THIS TOKEN AS THE 1ST *TOKEN* IN A  COMATOK  1528  A
COMATOK  *                      TOKEN CHAR MAP.                          COMATOK     1529     A
COMATOK  *                RMT = *RMT* BLOCK NAME                         COMATOK     1530     A
COMATOK  *                                                               COMATOK     1531     A
COMATOK  *         EXIT   NONE                                           COMATOK     1532     A
COMATOK  *                                                               COMATOK     1533     A
COMATOK  *         CALLS  NONE                                           COMATOK     1534     A
COMATOK                                                                  COMATOK     1535     A
COMATOK            PURGMAC '?TKMTOK                                      COMATOK     1536     A
COMATOK                                                                  COMATOK     1537     A
COMATOK            MACRO  '?TKMTOK,A,B,C,D,E                             COMATOK     1538     A
COMATOK  *                                                               COMATOK     1539     A
COMATOK   E RMT                                                          COMATOK     1540     A
COMATOK    IF -DEF,'?_B,1                                                COMATOK     1541     A
COMATOK   '?_B SET 0                                                     COMATOK     1542     A
COMATOK  *                                                               COMATOK     1543     A
COMATOK   D BSS 0                                                        COMATOK     1544     A
COMATOK  *                                                               COMATOK     1545     A
COMATOK    IF -DEF,"O="B,1                                               COMATOK     1546     A
COMATOK   "O="B BSS 0                                                    COMATOK     1547     A
COMATOK   '?TKS001 SET 60-TK.PADP-TK.PADL                                COMATOK     1548     A
COMATOK   A VFD '?TKS001/0,TK.PADL/C-FW.TOK,TK.LNKL/'?_B-FW.TOK,TK.TOTL/"O."B  COMATOK  1549  A
COMATOK   E RMT                                                          COMATOK     1550     A
COMATOK   '?TKMTOK ENDM                                                  COMATOK     1551     A
COMATOK   COMATOK  ENDX                                                  COMATOK     1552     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - COMATOK

          IDENTIFIER                TOTAL    ACTIVE

          COMATOK                    1552      1548
          CCGA015                       2         2
          CCGA022                       1         1

          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCBUB

COMCBUB  *COMDECK COMCBUB                                                COMCBUB        1     A
COMCBUB   BUB      CTEXT  COMCBUB - BURST/BUILD CHARACTERS WITH BLANK SQUEEZE.  COMCBUB  2  A
COMCBUB   BUB      SPACE  4,10                                           COMCBUB        3     A
COMCBUB            IF     -DEF,QUAL$,1                                   COMCBUB        4     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCBUB

```
COMCBUB            QUAL   COMCBUB                                                  COMCBUB      5    A
COMCBUB            BASE   D                                                        COMCBUB      6    A
COMCBUB   BUB      SPACE  4,10                                                     COMCBUB      7    A
COMCBUB  ***       BUB - BURST/BUILD CHARACTERS WITH BLANK SQUEEZE.                COMCBUB      8    A
COMCBUB  *         P CLARE  77/01/20                                               COMCBUB      9    A
COMCBUB   BUB      SPACE  4,10                                                     COMCBUB     10    A
COMCBUB  ***       BUB - BURST/BUILD CHARACTERS WITH BLANK SQUEEZE.                COMCBUB     11    A
COMCBUB  *                                                                         COMCBUB     12    A
COMCBUB  *                                                                         COMCBUB     13    A
COMCBUB  *         THIS ROUTINE, VIA A LEFT-TO-RIGHT SCAN, BURSTS CONSECUTIVE      COMCBUB     14    A
COMCBUB  *         CHARACTERS FROM A PACKED (10 CHARACTER PER WORD) SOURCE         COMCBUB     15    A
COMCBUB  *         LINE, AND BUILDS THEM INTO A SINGLE *GROUP* OF CHARACTERS.      COMCBUB     16    A
COMCBUB  *         IN ADDITION, *COMCBUB* IGNORES BLANKS.  NOTE THAT THE *GROUP*   COMCBUB     17    A
COMCBUB  *         THAT RESULTS FROM A SINGLE *BUB* CALL MAY BE MORE               COMCBUB     18    A
COMCBUB  *         THAN 1 CM WORD LONG, OR EVEN POSSIBLY OF ZERO LENGTH.           COMCBUB     19    A
COMCBUB  *                                                                         COMCBUB     20    A
COMCBUB  *         *COMCBUB* IS PRIMARILY INTENDED TO BE USED WITH *COMATOK*       COMCBUB     21    A
COMCBUB  *         AND *COMCTOK*, THE GENERAL PURPOSE ENTOKENING DISCIPLINE        COMCBUB     22    A
COMCBUB  *         (MORE LIKE SCHEME).  *COMCTOK* CALLS *BUB* TO PERFORM           COMCBUB     23    A
COMCBUB  *         EACH *TOGEL* /GROUP,...,SQZ/.                                   COMCBUB     24    A
COMCBUB  *                                                                         COMCBUB     25    A
COMCBUB  *         THE FORMAT OF THE RESULTING *GROUP* IS DEFINED AT               COMCBUB     26    A
COMCBUB  *         ASSEMBLY-TIME (DESCRIBED BELOW UNDER *ASSEMBLY-TIME OPTIONS*).  COMCBUB     27    A
COMCBUB  *                                                                         COMCBUB     28    A
COMCBUB  *         CHARACTERS IN THE INPUT SOURCE LINE ARE EXTRACTED FROM          COMCBUB     29    A
COMCBUB  *         LEFT-TO-RIGHT AND BUILT INTO THE RESULTANT *GROUP* BASED        COMCBUB     30    A
COMCBUB  *         ON A CHARACTER SHIFT MASK (IN (X1) ON ENTRY) THAT SPECIFIES     COMCBUB     31    A
COMCBUB  *         WHICH CHARACTERS CAN POTENTIALLY OCCUR IN THIS *GROUP*, I.E.    COMCBUB     32    A
COMCBUB  *         THE CHARACTER SHIFT MASK DEFINES WHICH CHARACTERS ARE           COMCBUB     33    A
COMCBUB  *         **ALLOWED** WITHIN THIS *GROUP*.                                COMCBUB     34    A
COMCBUB  *                                                                         COMCBUB     35    A
COMCBUB  *         (X1) HAS A BIT POSITION FOR EACH CHARACTER IN THE HOST          COMCBUB     36    A
COMCBUB  *         CHARACTER SET (CURRENTLY, ONLY DISPLAY CODE IS SUPPORTED.       COMCBUB     37    A
COMCBUB  *         *ASCII* SUPPORT HAS BEEN DESIGNED, BUT NOT IMPLEMENTED...YET).  COMCBUB     38    A
COMCBUB  *         *BUB* WILL BURST/BUILD ALL THE CHARACTERS THAT HAVE THEIR       COMCBUB     39    A
COMCBUB  *         RESPECTIVE BIT IN (X1) *ON* INTO A SINGLE *GROUP*.              COMCBUB     40    A
COMCBUB  *                                                                         COMCBUB     41    A
COMCBUB  *         IT WORKS LIKE THIS...  AS EACH CHARACTER IS BURST FROM THE      COMCBUB     42    A
COMCBUB  *         INPUT SOURCE LINE, ITS BINARY VALUE IS USED AS A LEFT-SHIFT     COMCBUB     43    A
COMCBUB  *         FOR (X1).  IF THE SHIFTED (X1) IS .MI., THEN THIS CHARACTER     COMCBUB     44    A
COMCBUB  *         IS BUILT INTO THE *GROUP* DEFINED BY (X1).  AND ALTERNATELY,    COMCBUB     45    A
COMCBUB  *         IF SHIFTED (X1) IS .PL., THEN THIS CHARACTER IS NOT IN THE      COMCBUB     46    A
COMCBUB  *         *GROUP* SO *BUB* RETURNS TO ITS CALLER.                         COMCBUB     47    A
COMCBUB  *                                                                         COMCBUB     48    A
COMCBUB  *         THE PROBLEM WITH MAPPING THE 64D DISPLAY CODE CHARACTERS,       COMCBUB     49    A
COMCBUB  *         ONE PER BIT, INTO A 60D BIT WORD IS THAT WE HAVE 4 CHARACTERS   COMCBUB     50    A
COMCBUB  *         TOO MANY.  BECAUSE THE LEFT-SHIFT HARDWARE SHIFTS MODULO 60D    COMCBUB     51    A
COMCBUB  *         (I.E. LX1 60D = LX1 00, LX1 61D = LX1 01, ETC), THIS MEANS       COMCBUB     52    A
COMCBUB  *         THAT 4 PAIRS OF CHARACTERS MAP INTO EACH OTHERS BITS.           COMCBUB     53    A
COMCBUB  *                                                                         COMCBUB     54    A
COMCBUB  *         I.E. --              : = @     00B = 60D OR 74B                COMCBUB     55    A
COMCBUB  *                              A = \     01B = 61D OR 75B                COMCBUB     56    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCBUB

```
COMCBUB  *                              B = ^     02B = 62D OR 76B                    COMCBUB      57     A
COMCBUB  *                              C = ;     03B = 63D OR 77B                    COMCBUB      58     A
COMCBUB  *                                                                            COMCBUB      59     A
COMCBUB  *         CONCEPTUALLY, THERE IS NOTHING WRONG WITH THIS AS LONG AS          COMCBUB      60     A
COMCBUB  *         THE PROGRAMMER DOESNT CARE WHETHER THESE CHARACTERS ARE            COMCBUB      61     A
COMCBUB  *         EQUIVALENT.  UNFORTUNATELY, FEW PEOPLE ARE WILLING TO              COMCBUB      62     A
COMCBUB  *         PUT UP WITH AN EQUALITY LIKE -A = @-.                              COMCBUB      63     A
COMCBUB  *                                                                            COMCBUB      64     A
COMCBUB  *         *BUB* CIRCUMVENTS THIS PROBLEM BY USING WHAT IS CALLED A           COMCBUB      65     A
COMCBUB  *         ROTATED CHARACTER SET, *RCS*, REPRESENTATION FOR DISPLAY           COMCBUB      66     A
COMCBUB  *         CODE.  THE TRICK IN THIS *RCS* TECHNIQUE IS TO TREAT THE           COMCBUB      67     A
COMCBUB  *         CHARACTER SET AS CIRCULAR, AND THEN TO ROTATE THE ENTIRE           COMCBUB      68     A
COMCBUB  *         CHARACTER SET SO AS TO FIND 4 PAIRS OF CONSECUTIVE CHARACTERS      COMCBUB      69     A
COMCBUB  *         THAT THE PROGRAMMER DOES NOT MIND BEING EQUIVALENT (FOR THE        COMCBUB      70     A
COMCBUB  *         PURPOSE OF *GROUP*ING).                                            COMCBUB      71     A
COMCBUB  *                                                                            COMCBUB      72     A
COMCBUB  *         FOR EXAMPLE, CONSIDER DISPLAY CODE --                              COMCBUB      73     A
COMCBUB  *                                                                            COMCBUB      74     A
COMCBUB  *                        ;    :                                             COMCBUB      75     A
COMCBUB  *                       77   00                                             COMCBUB      76     A
COMCBUB  *                \  76           01  A                                      COMCBUB      77     A
COMCBUB  *                    .           .                                          COMCBUB      78     A
COMCBUB  *                N  16           13  K                                      COMCBUB      79     A
COMCBUB  *                       15   14                                             COMCBUB      80     A
COMCBUB  *                        M    L                                             COMCBUB      81     A
COMCBUB  *                                                                            COMCBUB      82     A
COMCBUB  *         IT HAPPENS THAT  : = 00B, BUT THIS IS NOT **IMPERATIVE** AND       COMCBUB      83     A
COMCBUB  *         IS IN FACT QUITE INCONVENIENT DUE TO THE PROBLEM DESCRIBED         COMCBUB      84     A
COMCBUB  *         ABOVE.  WE COULD (AND DO) CHOOSE A DIFFERENT CHARACTER TO HAVE     COMCBUB      85     A
COMCBUB  *         THE VALUES 00B.  THE ONLY IMPORTANT THING IS THAT WE DO NOT        COMCBUB      86     A
COMCBUB  *         CHANGE THE ORDER OF CHARACTERS ON THE WHEEL.                       COMCBUB      87     A
COMCBUB  *                                                                            COMCBUB      88     A
COMCBUB  *         IF WE CHOOSE A CHARACTER IN THE MIDDLE OF THE ALPHABETICS          COMCBUB      89     A
COMCBUB  *         (-M-, FOR EXAMPLE), THEN ALL OUR TROUBLES VANISH INTO THIN         COMCBUB      90     A
COMCBUB  *         AIR --                                                             COMCBUB      91     A
COMCBUB  *                                                                            COMCBUB      92     A
COMCBUB  *                        L    M                                             COMCBUB      93     A
COMCBUB  *                       77   00                                             COMCBUB      94     A
COMCBUB  *                K  76           01  N                                      COMCBUB      95     A
COMCBUB  *                    .           .                                          COMCBUB      96     A
COMCBUB  *                    .           .                                          COMCBUB      97     A
COMCBUB  *                0  16           13  X                                      COMCBUB      98     A
COMCBUB  *                       15   14                                             COMCBUB      99     A
COMCBUB  *                        Z    Y                                             COMCBUB     100     A
COMCBUB  *                                                                            COMCBUB     101     A
COMCBUB  *         NOW THE OVERLAPPING CHARACTER PAIRS FALL IN THE MIDDLE OF THE      COMCBUB     102     A
COMCBUB  *         ALPHABETICS: I+M, J+N, K+O, L+P TO BE EXACT.                       COMCBUB     103     A
COMCBUB  *                                                                            COMCBUB     104     A
COMCBUB  *         THE CHARACTER THAT IS CHOSEN TO HAVE THE VALUE 00B IS CALLED       COMCBUB     105     A
COMCBUB  *         THE BIAS CHARACTER, *BICH*, AND IS DEFINED VIA AN ASSEMBLY         COMCBUB     106     A
COMCBUB  *         TIME OPTION (DEFAULT *BICH* = 1RM).                                COMCBUB     107     A
COMCBUB  *                                                                            COMCBUB     108     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCBUB

```
COMCBUB  *        USING THE DEFAULT *BICH*, CHARACTERS ARE MAPPED INTO (X1)          COMCBUB     109     A
COMCBUB  *        AS FOLLOWS --                                                      COMCBUB     110     A
COMCBUB  *                                                                           COMCBUB     111     A
COMCBUB  *        BIT   59 58 57 56 55 54 53 ... 47 46 45 44 43 ... 37 36 35 34      COMCBUB     112     A
COMCBUB  *        CHAR  IM JN KO LP  Q  R  S      Y  Z  0  1  2      8  9  +  -       COMCBUB     113     A
COMCBUB  *                                                                           COMCBUB     114     A
COMCBUB  *        BIT   33 32 ... 09 08 07 06 05 04 03 02 01 00                      COMCBUB     115     A
COMCBUB  *        CHAR   *  /      ;  :  A  B  C  D  E  F  G  H                       COMCBUB     116     A
COMCBUB  *                                                                           COMCBUB     117     A
COMCBUB  *        THE FORMULA FOR CONVERTING DISPLAY CODE (DPC) INTO THE ABOVE       COMCBUB     118     A
COMCBUB  *        *RCS* IS --                                                        COMCBUB     119     A
COMCBUB  *                                                                           COMCBUB     120     A
COMCBUB  *              RCS = DPC+10000B-BICH        (DEFAULT *BICH* = 1RM)          COMCBUB     121     A
COMCBUB  *                                                                           COMCBUB     122     A
COMCBUB  *        THE LOWER 6 BITS OF *RCS* SPECIFY THE LEFT SHIFT COUNT, *LS*,      COMCBUB     123     A
COMCBUB  *        FOR (X1).  IT CAN BE SEEN THAT THE CHARACTERS 'M' (LS=0) AND       COMCBUB     124     A
COMCBUB  *        'I' (LS=60D) MAP INTO THE SAME BIT IN (X1).  THIS IS ALSO TRUE     COMCBUB     125     A
COMCBUB  *        FOR THE CHARACTER PAIRS 'N' (LS=1) AND 'J' (LS=61D),               COMCBUB     126     A
COMCBUB  *        'O' (LS=2) AND 'K' (LS=62D), AND 'P' (LS=3) AND 'L' (LS=63D).      COMCBUB     127     A
COMCBUB  *                                                                           COMCBUB     128     A
COMCBUB  *                                                                           COMCBUB     129     A
COMCBUB  *        IN THE SECTIONS THAT FOLLOW, *ASSEMBLY-TIME DEPENDENCIES*          COMCBUB     130     A
COMCBUB  *        ARE DEFINED TO BE THE INTERFACES THAT THE *COMCBUB* INSTALLER      COMCBUB     131     A
COMCBUB  *        MUST PROVIDE.  *ASSEMBLY-TIME OPTIONS* ARE THOSE INTERFACES        COMCBUB     132     A
COMCBUB  *        WHICH ARE OPTIONAL, I.E. INTERFACES WHICH HAVE DEFAULT             COMCBUB     133     A
COMCBUB  *        CONDITIONS.                                                        COMCBUB     134     A
COMCBUB  *                                                                           COMCBUB     135     A
COMCBUB  *        ASSEMBLY-TIME DEPENDENCIES --                                      COMCBUB     136     A
COMCBUB  *                                                                           COMCBUB     137     A
COMCBUB  ************************************************************************   COMCBUB     138     A
COMCBUB  *                                                                           COMCBUB     139     A
COMCBUB  *        THE INSTALLER MUST PROVIDE THE FOLLOWING COMDECKS --               COMCBUB     140     A
COMCBUB  *              COMADEF   COMAREG                                            COMCBUB     141     A
COMCBUB  *                                                                           COMCBUB     142     A
COMCBUB  ************************************************************************   COMCBUB     143     A
COMCBUB  *                                                                           COMCBUB     144     A
COMCBUB  *        ASSEMBLY-TIME OPTIONS --                                           COMCBUB     145     A
COMCBUB  *                                                                           COMCBUB     146     A
COMCBUB  ************************************************************************   COMCBUB     147     A
COMCBUB  *                                                                           COMCBUB     148     A
COMCBUB  *        BICH - BIAS CHARACTER.           (DEFAULT =1RM)                    COMCBUB     149     A
COMCBUB  *                                                                           COMCBUB     150     A
COMCBUB  *        *BICH* SPECIFIES THE CHARACTER THAT IS TO OCCUPY BIT 59            COMCBUB     151     A
COMCBUB  *        OF (X1), I.E. THE CHARACTER THAT HAS AN *RCS* VALUE OF 00B.        COMCBUB     152     A
COMCBUB  *                                                                           COMCBUB     153     A
COMCBUB  ************************************************************************   COMCBUB     154     A
COMCBUB  *                                                                           COMCBUB     155     A
COMCBUB  *        *BB.XXXL* AND *BB.XXXP* ARE SYMBOLS THAT DEFINE A *COMADEF*        COMCBUB     156     A
COMCBUB  *        STRUCTURE THAT DESCRIBES THE FORM OF THE *GROUP*S THAT *BUB*       COMCBUB     157     A
COMCBUB  *        WILL GENERATE.  FOR EXAMPLE, SUPPOSE THE CALLING PROGRAM IS        COMCBUB     158     A
COMCBUB  *        USING *BUB* TO GENERATE TOKENS FOR INPUT SOURCE LINE               COMCBUB     159     A
COMCBUB  *        CHARACTERS.  THE *BB.* STRUCTURE WOULD DESCRIBE/DEFINE WHAT        COMCBUB     160     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCBUB

```
COMCBUB  *         HOST TOKENS LOOK LIKE.  SEE *COMADEF* AND *BB.* DEFINITIONS          COMCBUB    161    A
COMCBUB  *         BELOW.                                                               COMCBUB    162    A
COMCBUB  *                                                                              COMCBUB    163    A
COMCBUB  ************************************************************************      COMCBUB    164    A
COMCBUB   DEFS     SPACE  4,10                                                          COMCBUB    165    A
COMCBUB  **        MISCELLANEOUS SYMBOL DEFINITIONS.                                    COMCBUB    166    A
COMCBUB                                                                                 COMCBUB    167    A
COMCBUB                                                                                 COMCBUB    168    A
COMCBUB            IF     -DEF,BICH,1                                                   COMCBUB    169    A
COMCBUB   BICH     =      1RM                                                           COMCBUB    170    A
COMCBUB                                                                                 COMCBUB    171    A
COMCBUB            IF     -DEF,CHAR,1                                                   COMCBUB    172    A
COMCBUB   CHAR     =      6           NR OF BITS IN CHARACTER                           COMCBUB    173    A
COMCBUB   BB.      SPACE  4,10                                                          COMCBUB    174    A
COMCBUB  **        BB. - *BUB* GROUP STRUCTURE.                                         COMCBUB    175    A
COMCBUB  *                                                                              COMCBUB    176    A
COMCBUB  *         *BB.* SYMBOLS ARE OF THE FORM --                                     COMCBUB    177    A
COMCBUB  *                                                                              COMCBUB    178    A
COMCBUB  *                BB.XXXL  AND  BB.XXXP                                         COMCBUB    179    A
COMCBUB  *                                                                              COMCBUB    180    A
COMCBUB  *         WHERE FIELDS *XXX* ARE --                                            COMCBUB    181    A
COMCBUB  *                                                                              COMCBUB    182    A
COMCBUB  *         TOC  = TOKEN CHARACTERS.  *TOC* DESCRIBE/DEFINES THE                 COMCBUB    183    A
COMCBUB  *                LOCATION WITHIN PACKING REGISTER (X6) THAT *BUB*              COMCBUB    184    A
COMCBUB  *                IS TO PLACE *GROUP*ED CHARACTERS.  *BB.TOCL* IS               COMCBUB    185    A
COMCBUB  *                THE LENGTH (IN BITS) OF THIS FIELD, AND *BB.TOCP*             COMCBUB    186    A
COMCBUB  *                IS THE LOWEST (I.E. RIGHT-MOST) BIT POSITION IN THIS          COMCBUB    187    A
COMCBUB  *                FIELD.  DEFAULTS --  BB.TOCL = 60D, AND BB.TOCP = 00B.        COMCBUB    188    A
COMCBUB  *                                                                              COMCBUB    189    A
COMCBUB  *         TOT  = TOKEN TYPE.  EACH *GROUP* CAN BE ASSIGNED A TOKEN OR          COMCBUB    190    A
COMCBUB  *                *GROUP* TYPE AT EXECUTION TIME.  SEE (X6) ON ENTRY.           COMCBUB    191    A
COMCBUB  *                THIS FIELD DESCRIBE/DEFINES WHERE *BUB* IS TO PLACE           COMCBUB    192    A
COMCBUB  *                THE TOKEN TYPE IN PACKING REGISTER (X6).  THIS FIELD          COMCBUB    193    A
COMCBUB  *                IS NORMALLY THE COMPLEMENT TO *TOC*.  THAT IS,                COMCBUB    194    A
COMCBUB  *                BB.TOCL+BB.TOTL = 60D, I.E. A FULL WORD.                      COMCBUB    195    A
COMCBUB  *                DEFAULTS --  BB.TOTL = 0, AND BB.TOTP = 0.                    COMCBUB    196    A
COMCBUB                                                                                 COMCBUB    197    A
COMCBUB                                                                                 COMCBUB    198    A
COMCBUB   .BB      IF     -DEF,BB.TOCP                                                  COMCBUB    199    A
COMCBUB                                                                                 COMCBUB    200    A
COMCBUB            DESCRIBE BB.                                                         COMCBUB    201    A
COMCBUB   TOC      DEFINE 60                                                            COMCBUB    202    A
COMCBUB   TOT      DEFINE 0                                                             COMCBUB    203    A
COMCBUB   .BB      ENDIF                                                                COMCBUB    204    A
COMCBUB   BUB      SPACE  4,10                                                          COMCBUB    205    A
COMCBUB  **        BUB - BURST/BUILD WITH BLANK SQUEEZE.                                COMCBUB    206    A
COMCBUB  *                                                                              COMCBUB    207    A
COMCBUB  *                                                                              COMCBUB    208    A
COMCBUB  *         THE DISCUSSIONS OF *BUB*S ENTRY AND EXIT CONDITIONS ARE EACH         COMCBUB    209    A
COMCBUB  *         DIVIDED INTO 3 SECTIONS: MISCELLANEOUS, SOURCE INPUT, AND            COMCBUB    210    A
COMCBUB  *         *GROUP* OUTPUT.  THERE EXIST SUBTLETIES IN THESE ENTRY/EXIT          COMCBUB    211    A
COMCBUB  *         CONDITIONS THAT, ALTHOUGH HOPEFULLY NOT INCONSISTENT, DO             COMCBUB    212    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCBUB

```
COMCBUB  *        UNFORTUNATELY MAKE AN UNDERSTANDING OF HOW TO USE THIS          COMCBUB    213    A
COMCBUB  *        CHARACTER ACCESS METHOD NON-TRIVIAL.                            COMCBUB    214    A
COMCBUB  *                                                                        COMCBUB    215    A
COMCBUB  *        THIS DIFFICULTY ARISES SOLEY DUE TO EFFICIENCY CONSIDERATIONS.  COMCBUB    216    A
COMCBUB  *        *BUB* IS DESIGNED TO BE **VERY** FAST.  THIS UNFORTUNATELY      COMCBUB    217    A
COMCBUB  *        IMPLIES THAT SOME OF THE REGISTER CONVENTIONS ARE ORIENTED      COMCBUB    218    A
COMCBUB  *        MORE TOWARD THE INSTACK LOOP THAN TO A HUMAN.                   COMCBUB    219    A
COMCBUB  *                                                                        COMCBUB    220    A
COMCBUB  *        FOR EXAMPLE, (X4), THE REGISTER THAT IS USED WITHIN *BUB*       COMCBUB    221    A
COMCBUB  *        TO HOLD/MANIPULATE INDIVIDUAL CHARACTERS MAY, UPON ENTRY,       COMCBUB    222    A
COMCBUB  *        ALREADY CONTAIN A CHARACTER TO PROCESS.  THIS IS BECAUSE        COMCBUB    223    A
COMCBUB  *        *BUB* IS DESIGNED TO BE CALLED REPETITIVELY BY SOME EXECUTIVE   COMCBUB    224    A
COMCBUB  *        THAT IS PERFORMING A TRANSFORMATION UPON SOME CHARACTER         COMCBUB    225    A
COMCBUB  *        STRING (E.G. GENERATING TOKENS).                                COMCBUB    226    A
COMCBUB  *                                                                        COMCBUB    227    A
COMCBUB  *        IN THIS CIRCUMSTANCE, IT IS NOT DIFFICULT TO CONCEIVE OF SOME   COMCBUB    228    A
COMCBUB  *        OF *BUB*S EXIT CONDITIONS BEING USED AS *BUB*S ENTRY            COMCBUB    229    A
COMCBUB  *        CONDITIONS ON A SUBSEQUENT CALL.  THEREFORE, BECAUSE (X4)       COMCBUB    230    A
COMCBUB  *        USUALLY CONTAINS A CHARACTER ON EXIT FROM *BUB*, IT IS LIKELY   COMCBUB    231    A
COMCBUB  *        THAT *BUB* WILL BE CALLED AGAIN WITH (X4) AS AN ENTRY           COMCBUB    232    A
COMCBUB  *        CONDITION.  COMO SAVVY...                                       COMCBUB    233    A
COMCBUB  *                                                                        COMCBUB    234    A
COMCBUB  *        BY WAY OF INTRODUCTION, WE CAN MENTION THAT (A5,X5) CONTAIN     COMCBUB    235    A
COMCBUB  *        THE SOURCE INPUT CHARACTERS IN PACKED (10 CHAR PER WORD)        COMCBUB    236    A
COMCBUB  *        FORMAT AND (A6,X6) CONTAIN THE *GROUP* OUTPUT CHARACTERS        COMCBUB    237    A
COMCBUB  *        IN A FORMAT THAT IS DETERMINED BY THE *BB.XXXL* AND *BB.XXXP*   COMCBUB    238    A
COMCBUB  *        ASSEMBLY-TIME OPTIONS.                                          COMCBUB    239    A
COMCBUB  *                                                                        COMCBUB    240    A
COMCBUB  *                                                                        COMCBUB    241    A
COMCBUB  *        ENTRY  MISCELLANEOUS --                                         COMCBUB    242    A
COMCBUB  *                                                                        COMCBUB    243    A
COMCBUB  *              (B1) = 1                                                  COMCBUB    244    A
COMCBUB  *              (X0) = CHARACTER EXTRACT MASK (I.E. MX0 -CHAR).           COMCBUB    245    A
COMCBUB  *              (X1) = CHARACTER SHIFT MASK THAT DEFINES WHICH            COMCBUB    246    A
COMCBUB  *                     CHARACTERS CAN OCCUR IN THIS *GROUP*.  THAT IS,    COMCBUB    247    A
COMCBUB  *                     *BUB* WILL STOP IF IT ENCOUNTERS A SOURCE INPUT    COMCBUB    248    A
COMCBUB  *                     CHARACTER **NOT** SPECIFIED IN (X1).               COMCBUB    249    A
COMCBUB  *                                                                        COMCBUB    250    A
COMCBUB  *                     NOTE ALSO THAT (X1) **MUST** BE BIASED BY          COMCBUB    251    A
COMCBUB  *                     *BICH*  (SEE *ASSEMBLY-TIME OPTIONS*).  THIS       COMCBUB    252    A
COMCBUB  *                     MEANS THAT THE *BICH* BIAS CHARACTER OCCUPIES      COMCBUB    253    A
COMCBUB  *                     BIT 1S59 IN (X1).  SEE *COMCBUB* PREAMBLE.         COMCBUB    254    A
COMCBUB  *                                                                        COMCBUB    255    A
COMCBUB  *              SOURCE INPUT --                                           COMCBUB    256    A
COMCBUB  *                                                                        COMCBUB    257    A
COMCBUB  *              (X4) = 1ST SOURCE INPUT CHAR TO CHECK TO SEE IF IN THE    COMCBUB    258    A
COMCBUB  *                     *GROUP* DEFINED IN (X1), -R- FORMAT.  ELSE .MI.    COMCBUB    259    A
COMCBUB  *                     IF EMPTY, I.E. NEED TO EXTRACT A CHAR FROM (X5)    COMCBUB    260    A
COMCBUB  *                     FIRST.                                             COMCBUB    261    A
COMCBUB  *                                                                        COMCBUB    262    A
COMCBUB  *              (A5,X5) = A+C OF SOURCE WORD TO BEGIN BURST/BUILDING      COMCBUB    263    A
COMCBUB  *                     FROM.  (X5) CONTAINS THE SOURCE INPUT CHARACTERS   COMCBUB    264    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCBUB

```
COMCBUB  *                    IN PACKED (I.E. 10 CHARS PER WORD) FORMAT WITH          COMCBUB    265    A
COMCBUB  *                    THE 1ST CHAR TO EXTRACT IN THE HIGH ORDER CHAR          COMCBUB    266    A
COMCBUB  *                    POSITION (BITS 54 THRU 59).                             COMCBUB    267    A
COMCBUB  *                                                                            COMCBUB    268    A
COMCBUB  *                    NOTE THAT IF (X4) .PL., THEN (X5) IS NOT TOUCHED        COMCBUB    269    A
COMCBUB  *                    UNTIL (X4) HAS BEEN PROCESSED.                          COMCBUB    270    A
COMCBUB  *                                                                            COMCBUB    271    A
COMCBUB  *                    NOTE ALSO THAT IF (X5) IS EXHAUSTED DURING THIS         COMCBUB    272    A
COMCBUB  *                    *BUB* CALL, THEN (A5) IS INCREMENTED AND *BUB*          COMCBUB    273    A
COMCBUB  *                    CONTINUES BURST/BUILDING WITH FRESHLY LOADED            COMCBUB    274    A
COMCBUB  *                    (X5).                                                   COMCBUB    275    A
COMCBUB  *                                                                            COMCBUB    276    A
COMCBUB  *             (B6) = BIT COUNTER FOR (X5).  (B6) CONTAINS THE NR             COMCBUB    277    A
COMCBUB  *                    OF BITS IN (X5) REMAINING TO BU/BU.  IF (B6) IS         COMCBUB    278    A
COMCBUB  *                    OR BECOMES .LE. 0, THEN (X5) IS EMPTY (BY               COMCBUB    279    A
COMCBUB  *                    DEFINITION) AND THE NEXT SOURCE WORD IS LOADED          COMCBUB    280    A
COMCBUB  *                    INTO (A5,X5).                                           COMCBUB    281    A
COMCBUB  *                                                                            COMCBUB    282    A
COMCBUB  *                    IF (B6) IS 60D, THEN (X5) IS FULL.                      COMCBUB    283    A
COMCBUB  *                                                                            COMCBUB    284    A
COMCBUB  *             (B7) = SOURCE LINE REMAINING BIT COUNTER.                      COMCBUB    285    A
COMCBUB  *                    (B7) CONTAINS THE NR OF **BITS** REMAINING IN           COMCBUB    286    A
COMCBUB  *                    THE ENTIRE SOURCE LINE.  A BIT COUNT IS USED            COMCBUB    287    A
COMCBUB  *                    INSTEAD OF A WORD COUNT SO THAT *BUB* CAN BU/BU         COMCBUB    288    A
COMCBUB  *                    SOURCE LINES THAT DO NOT END ON A WORD BOUNDARY         COMCBUB    289    A
COMCBUB  *                    (A MOST COMMON OCCURENCE IN CDC-LAND).                  COMCBUB    290    A
COMCBUB  *                                                                            COMCBUB    291    A
COMCBUB  *                    NOTE THAT (B7) ARE **NOT** CHECKED UPON ENTRY,          COMCBUB    292    A
COMCBUB  *                    AND IN FACT ARE ONLY CHECKED WHEN (B6) INDICATE         COMCBUB    293    A
COMCBUB  *                    THAT (X5) IS EMPTY.  ONE CAN THINK OF THIS              COMCBUB    294    A
COMCBUB  *                    (B6)/(B7) RELATIONSHIP AS (B6) HAVING PRECEDENCE        COMCBUB    295    A
COMCBUB  *                    OVER (B7).                                              COMCBUB    296    A
COMCBUB  *                                                                            COMCBUB    297    A
COMCBUB  *                    ALTHOUGH THIS IS PERHAPS NOT THE MOST HUMAN             COMCBUB    298    A
COMCBUB  *                    REGISTER RELATIONSHIP, IT IS THE MOST SPEED             COMCBUB    299    A
COMCBUB  *                    EFFICIENT ONE FOR *BUB*S INSTACK LOOP.                  COMCBUB    300    A
COMCBUB  *                                                                            COMCBUB    301    A
COMCBUB  *             *GROUP* OUTPUT --                                              COMCBUB    302    A
COMCBUB  *                                                                            COMCBUB    303    A
COMCBUB  *             (X6) = PACKING REGISTER FOR *GROUP* OUTPUT WORDS.              COMCBUB    304    A
COMCBUB  *                    *BUB* WILL CONTINUE TO PACK SOURCE INPUT                COMCBUB    305    A
COMCBUB  *                    CHARACTERS TO (X6) AS LONG AS HE KEEPS                  COMCBUB    306    A
COMCBUB  *                    ENCOUNTERING CHARACTERS THAT HAVE BEEN SPECIFIED        COMCBUB    307    A
COMCBUB  *                    VIA (X1) .AND. AS LONG AS (X6) IS NOT FULL.             COMCBUB    308    A
COMCBUB  *                                                                            COMCBUB    309    A
COMCBUB  *                    UPON ENTRY, (B3) SPECIFIES THE LEFT-SHIFT COUNT         COMCBUB    310    A
COMCBUB  *                    FOR THE NEXT CHARACTER IN (X4) THAT *BUB* IS            COMCBUB    311    A
COMCBUB  *                    TO PACK INTO (X6).  REMEMBER THAT A CHARACTER           COMCBUB    312    A
COMCBUB  *                    IN (X4) IS IN -R- FMT.  I.E. --                         COMCBUB    313    A
COMCBUB  *                                                                            COMCBUB    314    A
COMCBUB  *                        BX4  -X0*X5      (X4) = CHAR, IN -R- FMT            COMCBUB    315    A
COMCBUB  *                        LX3  X4,B3                                          COMCBUB    316    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCBUB

```
COMCBUB  *                          BX6  X6+X3       MERGE (X4) INTO (X6)
COMCBUB  *
COMCBUB  *                    WILL PACK A CHARACTER TO (X6).
COMCBUB  *
COMCBUB  *                    CHARACTERS ARE PACKED INTO (X6) FROM LEFT-TO-
COMCBUB  *                    RIGHT, BEGINNING AT LEFT-MOST CHARACTER POSITION
COMCBUB  *                    IN *BB.TOCP/BB.TOCL* FIELD.  (X6) IS **FULL**,
COMCBUB  *                    BY DEFINITION, WHEN THIS *BB.TOCP/BB.TOCL* FIELD
COMCBUB  *                    IS **FULL** (I.E. WHEN (B3) .LE. *BB.TOCP*).
COMCBUB  *
COMCBUB  *                    WHEN THIS OCCURS, (X6) WILL BE STORED BY
COMCBUB  *                    INCREMENTING (A6).  THE *BB.TOCP/BB.TOCL* FIELD
COMCBUB  *                    IN (X6) IS THEN CLEARED AND BURST/BUILDING
COMCBUB  *                    CONTINUES.  NOTE THAT *BUB* **NEVER** TOUCHES
COMCBUB  *                    ANYTHING BUT *BB.TOCP/BB.TOCL* FIELD IN (X6).
COMCBUB  *
COMCBUB  *                    IN THIS WAY THEN, (X6) UPON ENTRY, MAY CONTAIN
COMCBUB  *                    A TOKEN OR GROUP TYPE IN *BB.TOTP/BB.TOTL*
COMCBUB  *                    FIELD AND THIS TOKEN TYPE WILL BE PROPAGATED
COMCBUB  *                    INTO EVERY TOKEN *BUB* BUILDS INTO (X6).
COMCBUB  *
COMCBUB  *                                ** LIMITATION **
COMCBUB  *
COMCBUB  *                    (X6) MUST NOT BE FULL UPON ENTRY (I.E. (B3) .LT.
COMCBUB  *                    *BB.TOCP*), AS *BUB* PACKS A CHARACTER TO (X6)
COMCBUB  *                    **BEFORE** CHECKING FOR *(X6) FULL*.
COMCBUB  *
COMCBUB  *                                   * * *
COMCBUB  *
COMCBUB  *                    NOTE THAT IF (X6) IS LEFT PARTIALLY FILLED WHEN
COMCBUB  *                    *BUB* SENSES AN EXITING CONDITION, THEN (X6) IS
COMCBUB  *                    STORED OUT BEFORE EXITING.
COMCBUB  *
COMCBUB  *              (A6)+1 = ADDR TO STORE THE NEXT *GROUP* OUTPUT WORD
COMCBUB  *                    (OR TOKEN) GENERATED BY THIS *BUB* CALL,
COMCBUB  *                    I.E. (X6) IS STORED OUT BY INCREMENTING (A6).
COMCBUB  *
COMCBUB  *              (B3) = LEFT-SHIFT COUNT FOR 1ST CHARACTER *BUB* BUILDS
COMCBUB  *                    INTO (X6).  I.E. IF (X6) IS TO BE CONSIDERED
COMCBUB  *                    EMPTY UPON ENTRY, THEN --
COMCBUB  *
COMCBUB  *                        (B3) = BB.TOCP+BB.TOCL-CHAR
COMCBUB  *
COMCBUB  *                    WHERE *BB.TOCP* = RIGHT-MOST BIT POSITION THAT
COMCBUB  *                                      *BUB* CAN BUILD CHARACTERS
COMCBUB  *                                      INTO (X6).
COMCBUB  *                         *BB.TOCL* = LEN (IN BITS) OF FIELD WITHIN
COMCBUB  *                                      (X6) TO BUILD CHARACTERS.
COMCBUB  *                         *CHAR*    = LEN (IN BITS) OF A SINGLE
COMCBUB  *                                      CHARACTER.  CURRENTLY ONLY
COMCBUB  *                                      6 BIT OR LESS SUPPORTED.
COMCBUB  *

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCBUB

```
COMCBUB  *         EXIT   (B1) = 1                                          COMCBUB    369    A
COMCBUB  *                (X0) = AS ON ENTRY.                               COMCBUB    370    A
COMCBUB  *                (X1) = AS ON ENTRY.                               COMCBUB    371    A
COMCBUB  *                (X4) = CHAR THAT TERMINATED THIS *GROUP*, IN -R- FMT.  COMCBUB    372    A
COMCBUB  *                       I.E. 1ST CHAR ENCOUNTERED THAT WAS NOT CHARACTER  COMCBUB    373    A
COMCBUB  *                       RANGE SPECIFIED VIA (X1).                  COMCBUB    374    A
COMCBUB  *                (A5,X5) = A+C OF CURRENT SOURCE WORD.  (X5) IS SHIFTED  COMCBUB    375    A
COMCBUB  *                          SO THAT NEXT CHAR TO EXTRACT IS IN LEFT-MOST  COMCBUB    376    A
COMCBUB  *                          CHARACTER POSITION (I.E. BITS 59 THRU 54 FOR  COMCBUB    377    A
COMCBUB  *                          DISPLAY CODE, 6 BIT).  THEREFORE, (X4) IS  COMCBUB    378    A
COMCBUB  *                          IN BITS 5 THRU 0.                         COMCBUB    379    A
COMCBUB  *                (B6) = NR OF BITS REMAINING TO EXTRACT FROM (X5).  COMCBUB    380    A
COMCBUB  *                       I.E. DOES **NOT** INCLUDE CHAR IN (X4).    COMCBUB    381    A
COMCBUB  *                (B7) = .GE. 0 IF NR OF BITS REMAINING TO BURST/BUILD  COMCBUB    382    A
COMCBUB  *                       IN THIS SOURCE LINE.  ELSE .LT. 0 IF DONE.  COMCBUB    383    A
COMCBUB  *                (A6) = ADDR OF MOST RECENT TOKEN GENERATED.       COMCBUB    384    A
COMCBUB  *                (X6) = PACKING REGISTER.  (X6) CAN ONLY BE DETERMINED  COMCBUB    385    A
COMCBUB  *                       BY QUERYING (B3).                          COMCBUB    386    A
COMCBUB  *                (B3) = LEFT-SHIFT COUNT FOR NEXT CHARACTER THAT WOULD  COMCBUB    387    A
COMCBUB  *                       HAVE (BUT WASNT) GENERATED TO (X6).        COMCBUB    388    A
COMCBUB  *                                                                  COMCBUB    389    A
COMCBUB  *                       THEREFORE, IF (B3) = BB.TOCP+BB.TOCL-CHAR  COMCBUB    390    A
COMCBUB  *                       THEN *BB.TOCP/BB.TOCL* FIELD IN (X6) IS EMPTY.  COMCBUB    391    A
COMCBUB  *                       ELSE IF (B3) .GE. BB.TOCP, THEN (X6) CONTAINS  COMCBUB    392    A
COMCBUB  *                       MOST RECENT TOKEN BUILT AND STORED AT (A6).  COMCBUB    393    A
COMCBUB  *                                                                  COMCBUB    394    A
COMCBUB  *         USES   X - 2,3,4,5,6,7                                   COMCBUB    395    A
COMCBUB  *                A - 2,3,5,6                                       COMCBUB    396    A
COMCBUB  *                B - 2,3,4,5,6,7                                   COMCBUB    397    A
COMCBUB  *                                                                  COMCBUB    398    A
COMCBUB  *         CALLS  NONE                                              COMCBUB    399    A
COMCBUB                                                                     COMCBUB    400    A
COMCBUB                                                                     COMCBUB    401    A
COMCBUB   BUB      SUBR                ** ENTRY/EXIT **                     COMCBUB    402    A
COMCBUB                                                                     COMCBUB    403    A
COMCBUB  *         INITIALIZATION.                                          COMCBUB    404    A
COMCBUB                                                                     COMCBUB    405    A
COMCBUB            SA3    60D         (A3) = NR OF BITS IN A WORD (CONSTANT)  COMCBUB    406    A
COMCBUB            SA2    =10H                                              COMCBUB    407    A
COMCBUB            SB4    BB.TOCP     (B4) = LOWEST BIT POSITION THAT WE CAN  COMCBUB    408    A
COMCBUB  *                                   SHIFT CHARS INTO (X6)          COMCBUB    409    A
COMCBUB            SB5    1R +10000B-BICH                                   COMCBUB    410    A
COMCBUB            SB7    A3-B7       (B7) = -(BIT COUNT - 60D)             COMCBUB    411    A
COMCBUB            MI     X4,BUB3     IF (X4) DOES NOT CONTAIN A USEABLE CHAR  COMCBUB    412    A
COMCBUB            SB6    B6+CHAR                                           COMCBUB    413    A
COMCBUB            EQ     BUB4                                              COMCBUB    414    A
COMCBUB                                                                     COMCBUB    415    A
COMCBUB  *         BURST/BUILD LOOP  ** INSTACK **                          COMCBUB    416    A
COMCBUB                                                                     COMCBUB    417    A
COMCBUB   BUB2     SB7    B7+A3                                             COMCBUB    418    A
COMCBUB            SA5    A5+B1                                             COMCBUB    419    A
COMCBUB            GE     B7,B0,BUB5  IF NO MORE SOURCE WORDS TO BU/BU      COMCBUB    420    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCBUB

```
COMCBUB            IX7    X5-X2                                                  COMCBUB    421    A
COMCBUB            SB6    A3                                                     COMCBUB    422    A
COMCBUB            ZR     X7,BUB2     IF NEXT SOURCE WORD IS ALL BLANK, IGNORE...COMCBUB    423    A
COMCBUB                                                                           COMCBUB    424    A
COMCBUB   BUB3     LX5    CHAR                                                   COMCBUB    425    A
COMCBUB            LE     B6,B0,BUB2  IF SOURCE WORD (X5) IS EXHAUSTED           COMCBUB    426    A
COMCBUB            BX4    -X0*X5                                                 COMCBUB    427    A
COMCBUB                                                                           COMCBUB    428    A
COMCBUB   BUB4     SB6    B6-CHAR                                                COMCBUB    429    A
COMCBUB            SB2    X4+10000B-BICH                                         COMCBUB    430    A
COMCBUB            LX7    X1,B2                                                  COMCBUB    431    A
COMCBUB            EQ     B2,B5,BUB3  IF NEXT CHAR TO PACK IS A BLANK (55B)      COMCBUB    432    A
COMCBUB            LX3    X4,B3                                                  COMCBUB    433    A
COMCBUB            PL     X7,BUB6     IF NEXT CHAR TO PACK NOT IN THIS GROUP     COMCBUB    434    A
COMCBUB            SB3    B3-CHAR                                                COMCBUB    435    A
COMCBUB            BX6    X6+X3                                                  COMCBUB    436    A
COMCBUB            GE     B3,B4,BUB3  IF PACKING REGISTER (X6) NOT FULL          COMCBUB    437    A
COMCBUB                                                                           COMCBUB    438    A
COMCBUB   *        HERE IF PACKING REGISTER (X6) IS FULL.                         COMCBUB    439    A
COMCBUB                                                                           COMCBUB    440    A
COMCBUB            MX3    -BB.TOTL                                               COMCBUB    441    A
COMCBUB            SA6    A6+B1                                                  COMCBUB    442    A
COMCBUB            SB3    BB.TOCP+BB.TOCL-CHAR                                   COMCBUB    443    A
COMCBUB            LX3    BB.TOTP-0                                              COMCBUB    444    A
COMCBUB            BX6    -X3*X6      CLEAR *TOC*                                COMCBUB    445    A
COMCBUB            EQ     BUB3        BACK INTO THE FIRE...                      COMCBUB    446    A
COMCBUB                                                                           COMCBUB    447    A
COMCBUB   *        HERE AT LAST/FINAL WORD.                                       COMCBUB    448    A
COMCBUB   *                                                                       COMCBUB    449    A
COMCBUB   *          1. IF HAVE NOT PROCESSED LAST/FINAL WORD YET, THEN WE NEED   COMCBUB    450    A
COMCBUB   *             TO GO BACK AND BURST/BUILD IT.                            COMCBUB    451    A
COMCBUB   *                                                                       COMCBUB    452    A
COMCBUB   *          2. IF HAVE ALREADY BU/BU THE LAST/FINAL WORD, THEN DONE...   COMCBUB    453    A
COMCBUB                                                                           COMCBUB    454    A
COMCBUB   BUB5     SB6    A3-B7                                                  COMCBUB    455    A
COMCBUB            GT     B6,B0,BUB3  IF NEED TO PROCESS FINAL SOURCE WORD       COMCBUB    456    A
COMCBUB                                                                           COMCBUB    457    A
COMCBUB   *        FINAL PROCESSING...(X6) MIGHT NEED TO BE STORED OUT.           COMCBUB    458    A
COMCBUB                                                                           COMCBUB    459    A
COMCBUB   BUB6     SB4    B3-BB.TOCP-BB.TOCL+CHAR                                COMCBUB    460    A
COMCBUB            SB7    A3-B7       RESTORE (B7) TO A CIVIL FORM               COMCBUB    461    A
COMCBUB            ZR     B4,EXIT.    IF (X6) IS EMPTY, DONE...                  COMCBUB    462    A
COMCBUB            SA6    A6+1                                                   COMCBUB    463    A
COMCBUB            EQ     EXIT.                                                  COMCBUB    464    A
COMCBUB   BUB      SPACE  4,10                                                   COMCBUB    465    A
COMCBUB            BASE   *                                                      COMCBUB    466    A
COMCBUB   QUAL$    IF     -DEF,QUAL$                                             COMCBUB    467    A
COMCBUB            QUAL   *                                                      COMCBUB    468    A
COMCBUB   BUB      =      /COMCBUB/BUB                                           COMCBUB    469    A
COMCBUB   QUAL$    ENDIF                                                         COMCBUB    470    A
COMCBUB   COMCBUB  ENDX                                                          COMCBUB    471    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - COMCBUB

```
                IDENTIFIER                TOTAL    ACTIVE

                COMCBUB                     471       471

                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCBUN

COMCBUN  *COMDECK COMCBUN                                                    COMCBUN        1     A
COMCBUN   BUN      CTEXT  COMCBUN - BURST/BUILD CHARACTERS WITH NO BLANK SQUEEZE.  COMCBUN        2     A
COMCBUN   BUN      SPACE  4,10                                               COMCBUN        3     A
COMCBUN            IF     -DEF,QUAL$,1                                       COMCBUN        4     A
COMCBUN            QUAL   COMCBUN                                            COMCBUN        5     A
COMCBUN            BASE   D                                                  COMCBUN        6     A
COMCBUN   BUN      SPACE  4,10                                               COMCBUN        7     A
COMCBUN  ***       BUN - BURST/BUILD CHARACTERS WITH NO BLANK SQUEEZE.       COMCBUN        8     A
COMCBUN  *         P CLARE  77/01/20                                         COMCBUN        9     A
COMCBUN   BUN      SPACE  4,10                                               COMCBUN       10     A
COMCBUN  ***       BUN - BURST/BUILD CHARACTERS WITH NO BLANK SQUEEZE.       COMCBUN       11     A
COMCBUN  *                                                                   COMCBUN       12     A
COMCBUN  *                                                                   COMCBUN       13     A
COMCBUN  *         THIS ROUTINE, VIA A LEFT-TO-RIGHT SCAN, BURSTS CONSECUTIVE  COMCBUN       14     A
COMCBUN  *         CHARACTERS FROM A PACKED (10 CHARACTER PER WORD) SOURCE   COMCBUN       15     A
COMCBUN  *         LINE, AND BUILDS THEM INTO A SINGLE *GROUP* OF CHARACTERS.  COMCBUN       16     A
COMCBUN  *         UNLIKE *COMCBUB*, *BUN* IGNORES NO CHARACTERS (I.E. BLANKS  COMCBUN       17     A
COMCBUN  *         ARE NOT SQUEEZED).                                        COMCBUN       18     A
COMCBUN  *                                                                   COMCBUN       19     A
COMCBUN  *         *COMCBUN* IS PRIMARILY INTENDED TO BE USED WITH *COMATOK* COMCBUN       20     A
COMCBUN  *         AND *COMCTOK*, THE GENERAL PURPOSE ENTOKENING DISCIPLINE. COMCBUN       21     A
COMCBUN  *         *COMCTOK* CALLS *BUN* TO PERFORM EACH                     COMCBUN       22     A
COMCBUN  *         *TOGEL* /GROUP,...,NSQZ/.                                 COMCBUN       23     A
COMCBUN  *                                                                   COMCBUN       24     A
COMCBUN  *         IN THE INTEREST OF BREVITY, THE DESCRIPTION OF HOW THIS   COMCBUN       25     A
COMCBUN  *         ROUTINE WORKS HAS BEEN OMITTED.  IT CAN BE FOUND, HOWEVER, COMCBUN       26     A
COMCBUN  *         WITH *COMCBUB* AS THESE 2 ROUTINES WORK MUCH IN THE SAME  COMCBUN       27     A
COMCBUN  *         WAY.  I ALSO SUGGEST READING THE FTN 5 IMS FOR THE DECK *LEX*, COMCBUN       28     A
COMCBUN  *         BECAUSE THAT PRESENTATION TENDS TO BE MORE LEISURELY      COMCBUN       29     A
COMCBUN  *         AND EASIER TO FOLLOW...                                   COMCBUN       30     A
COMCBUN  *                                                                   COMCBUN       31     A
COMCBUN  *         IN THE SECTIONS THAT FOLLOW, *ASSEMBLY-TIME DEPENDENCIES* COMCBUN       32     A
COMCBUN  *         ARE DEFINED TO BE THE INTERFACES THAT THE *COMCBUN* INSTALLER  COMCBUN       33     A
COMCBUN  *         MUST PROVIDE.  *ASSEMBLY-TIME OPTIONS* ARE THOSE INTERFACES  COMCBUN       34     A
COMCBUN  *         WHICH ARE OPTIONAL, I.E. INTERFACES WHICH HAVE DEFAULT    COMCBUN       35     A
COMCBUN  *         CONDITIONS.                                               COMCBUN       36     A
COMCBUN  *                                                                   COMCBUN       37     A
COMCBUN  *         ASSEMBLY-TIME DEPENDENCIES --                             COMCBUN       38     A
COMCBUN  *                                                                   COMCBUN       39     A
COMCBUN  ***********************************************************************  COMCBUN       40     A
COMCBUN  *                                                                   COMCBUN       41     A
COMCBUN  *         THE INSTALLER MUST PROVIDE THE FOLLOWING COMDECKS --      COMCBUN       42     A
COMCBUN  *                COMADEF   COMAREG                                  COMCBUN       43     A
COMCBUN  *                                                                   COMCBUN       44     A
COMCBUN  ***********************************************************************  COMCBUN       45     A
COMCBUN  *                                                                   COMCBUN       46     A

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCBUN

```
COMCBUN  *         ASSEMBLY-TIME OPTIONS --                                COMCBUN    47    A
COMCBUN  *                                                                 COMCBUN    48    A
COMCBUN  ************************************************************      COMCBUN    49    A
COMCBUN  *                                                                 COMCBUN    50    A
COMCBUN  *         BICH - BIAS CHARACTER.            (DEFAULT =1RM)       COMCBUN    51    A
COMCBUN  *                                                                 COMCBUN    52    A
COMCBUN  *         *BICH* SPECIFIES THE CHARACTER THAT IS TO OCCUPY BIT 59 COMCBUN    53    A
COMCBUN  *         OF (X1), I.E. THE CHARACTER THAT HAS AN *RCS* VALUE OF 00B.  COMCBUN  54  A
COMCBUN  *                                                                 COMCBUN    55    A
COMCBUN  ************************************************************      COMCBUN    56    A
COMCBUN  *                                                                 COMCBUN    57    A
COMCBUN  *         *BB.XXXL* AND *BB.XXXP* ARE SYMBOLS THAT DEFINE A *COMADEF*  COMCBUN  58  A
COMCBUN  *         STRUCTURE THAT DESCRIBES THE FORM OF THE *GROUP*S THAT *BUN*  COMCBUN  59  A
COMCBUN  *         WILL GENERATE.  FOR EXAMPLE, SUPPOSE THE CALLING PROGRAM IS  COMCBUN  60  A
COMCBUN  *         USING *BUN* TO GENERATE TOKENS FOR INPUT SOURCE LINE  COMCBUN  61  A
COMCBUN  *         CHARACTERS.  THE *BB.* STRUCTURE WOULD DESCRIBE/DEFINE WHAT  COMCBUN  62  A
COMCBUN  *         HOST TOKENS LOOK LIKE.  SEE *COMADEF* AND *BB.* DEFINITIONS  COMCBUN  63  A
COMCBUN  *         BELOW.                                                  COMCBUN    64    A
COMCBUN  *                                                                 COMCBUN    65    A
COMCBUN  ************************************************************      COMCBUN    66    A
COMCBUN   DEFS     SPACE  4,10                                             COMCBUN    67    A
COMCBUN  **        MISCELLANEOUS SYMBOL DEFINITIONS.                       COMCBUN    68    A
COMCBUN                                                                    COMCBUN    69    A
COMCBUN                                                                    COMCBUN    70    A
COMCBUN            IF     -DEF,BICH,1                                      COMCBUN    71    A
COMCBUN   BICH     =      1RM                                              COMCBUN    72    A
COMCBUN                                                                    COMCBUN    73    A
COMCBUN            IF     -DEF,CHAR,1                                      COMCBUN    74    A
COMCBUN   CHAR     =      6           NR OF BITS IN CHARACTER              COMCBUN    75    A
COMCBUN   BB.      SPACE  4,10                                             COMCBUN    76    A
COMCBUN  **        BB. - *BUN* GROUP STRUCTURE.                            COMCBUN    77    A
COMCBUN  *                                                                 COMCBUN    78    A
COMCBUN  *         *BB.* SYMBOLS ARE OF THE FORM --                        COMCBUN    79    A
COMCBUN  *                                                                 COMCBUN    80    A
COMCBUN  *                BB.XXXL  AND  BB.XXXP                            COMCBUN    81    A
COMCBUN  *                                                                 COMCBUN    82    A
COMCBUN  *         WHERE FIELDS *XXX* ARE --                               COMCBUN    83    A
COMCBUN  *                                                                 COMCBUN    84    A
COMCBUN  *         TOC  = TOKEN CHARACTERS.  *TOC* DESCRIBE/DEFINES THE    COMCBUN    85    A
COMCBUN  *                LOCATION WITHIN PACKING REGISTER (X6) THAT *BUN*  COMCBUN  86  A
COMCBUN  *                IS TO PLACE *GROUP*ED CHARACTERS.  *BB.TOCL* IS  COMCBUN  87  A
COMCBUN  *                THE LENGTH (IN BITS) OF THIS FIELD, AND *BB.TOCP*  COMCBUN  88  A
COMCBUN  *                IS THE LOWEST (I.E. RIGHT-MOST) BITS POSITION IN THIS  COMCBUN  89  A
COMCBUN  *                FIELD.  DEFAULTS --  BB.TOCL = 60D, AND BB.TOCP = 00B.  COMCBUN  90  A
COMCBUN  *                                                                 COMCBUN    91    A
COMCBUN  *         TOT  = TOKEN TYPE.  EACH *GROUP* CAN BE ASSIGNED A TOKEN OR  COMCBUN  92  A
COMCBUN  *                *GROUP* TYPE AT EXECUTION TIME.  SEE (X6) ON ENTRY.  COMCBUN  93  A
COMCBUN  *                THIS FIELD DESCRIBE/DEFINES WHERE *BUN* IS TO PLACE  COMCBUN  94  A
COMCBUN  *                THE TOKEN TYPE IN PACKING REGISTER (X6).  THIS FIELD  COMCBUN  95  A
COMCBUN  *                IS NORMALLY THE COMPLEMENT TO *TOC*.  THAT IS,   COMCBUN  96  A
COMCBUN  *                BN.TOCL+BN.TOTL = 60D, I.E. A FULL WORD.         COMCBUN  97  A
COMCBUN  *                DEFAULTS --  BN.TOTL = 0, AND BN.TOTP = 0.       COMCBUN  98  A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

*** OUTPUT LINE LIMIT EXCEEDED. LIST OPTIONS 3 AND 4 CANCELLED. ***

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCBUN

```
COMCBUN                                                                  COMCBUN      99     A
COMCBUN                                                                  COMCBUN     100     A
COMCBUN   .BN      IF     -DEF,BN.TOCP                                   COMCBUN     101     A
COMCBUN                                                                  COMCBUN     102     A
COMCBUN            DESCRIBE BN.                                          COMCBUN     103     A
COMCBUN   TOC      DEFINE 60                                             COMCBUN     104     A
COMCBUN   TOT      DEFINE 0                                              COMCBUN     105     A
COMCBUN   .BN      ENDIF                                                 COMCBUN     106     A
COMCBUN   BUN      SPACE  4,10                                           COMCBUN     107     A
COMCBUN  **        BUN - BURST/BUILD WITH NO BLANK SQUEEZE.              COMCBUN     108     A
COMCBUN  *                                                               COMCBUN     109     A
COMCBUN  *                                                               COMCBUN     110     A
COMCBUN  *         ENTRY  (B1) = 1                                       COMCBUN     111     A
COMCBUN  *                (A5) = ADDR OF 1ST WORD TO BURST/BUILD         COMCBUN     112     A
COMCBUN  *                (X5) = 1ST WORD TO BURST/BUILD, CONTAINING CHARS COMCBUN   113     A
COMCBUN  *                       TO BURST/BUILD IN PACKED (10 CHAR/WORD) FMT. COMCBUN 114     A
COMCBUN  *                       (X5) IS SHIFTED SO THAT 1ST CHAR TO BU/BU IS COMCBUN 115     A
COMCBUN  *                       IN BITS 5 THRU 0.                       COMCBUN     116     A
COMCBUN  *                (B6) = NR OF BITS IN (X5) REMAINING TO BU/BU.  COMCBUN     117     A
COMCBUN  *                       IF (B6) .LE. 0, THEN (X5) IS EMPTY.  IN THIS COMCBUN 118     A
COMCBUN  *                       CASE, THE NEXT SOURCE LINE IMAGE WORD AT COMCBUN 119     A
COMCBUN  *                       (A5)+1 WILL BE LOADED INTO (X5) BEFORE  COMCBUN     120     A
COMCBUN  *                       PROCEEDING.  IF (B6) = 60D, THEN (X5) IS FULL. COMCBUN 121 A
COMCBUN  *                (B7) = NR OF SOURCE LINE IMAGE WORDS REMAINING TO BE COMCBUN 122 A
COMCBUN  *                       BU/BU.  IF (B7) .LE. 0 UPON ENTRY .AND. (B6) COMCBUN 123 A
COMCBUN  *                       INDICATES THAT (X5) NOT EMPTY, THEN *BUN* WILL COMCBUN 124 A
COMCBUN  *                       BU/BU (X5) BEFORE CHECKING (B7).        COMCBUN     125     A
COMCBUN  *                (A6)+1 = ADDR TO STORE 1ST TOKEN GENERATED     COMCBUN     126     A
COMCBUN  *                (X6) = PACKING REGISTER FOR BUILDING TOKENS.   COMCBUN     127     A
COMCBUN  *                                                               COMCBUN     128     A
COMCBUN  *         EXIT   (B1) = 1                                       COMCBUN     129     A
COMCBUN  *                (A5,X5),(B6),(B7),(A6),(B3) ADJUSTED IF NECESSARY COMCBUN 130 A
COMCBUN  *                                                               COMCBUN     131     A
COMCBUN  *         USES   ALL BUT (A0)                                   COMCBUN     132     A
COMCBUN  *                                                               COMCBUN     133     A
COMCBUN  *         CALLS  NONE                                           COMCBUN     134     A
COMCBUN                                                                  COMCBUN     135     A
COMCBUN                                                                  COMCBUN     136     A
COMCBUN   BUN      SUBR                ** ENTRY/EXIT **                  COMCBUN     137     A
COMCBUN                                                                  COMCBUN     138     A
COMCBUN  *         INITIALIZATION.                                       COMCBUN     139     A
COMCBUN                                                                  COMCBUN     140     A
COMCBUN            SA3    60D          (A3) = NR OF BITS IN A WORD (CONSTANT) COMCBUN 141 A
COMCBUN            SB4    BN.TOCP      (B4) = LOWEST BIT POSITION THAT WE CAN COMCBUN 142 A
COMCBUN  *                                   SHIFT CHARS INTO (X6)       COMCBUN     143     A
COMCBUN            SB7    A3-B7        (B7) = -(BIT COUNT-60)              COMCBUN     144     A
COMCBUN            MI     X4,BUN3      IF (X4) DOES NOT CONTAIN A USEABLE CHAR COMCBUN 145 A
COMCBUN            SB6    B6+CHAR                                         COMCBUN     146     A
COMCBUN            EQ     BUN4                                            COMCBUN     147     A
COMCBUN                                                                  COMCBUN     148     A
COMCBUN  *         BURST/BUILD LOOP  ** INSTACK **                       COMCBUN     149     A
COMCBUN                                                                  COMCBUN     150     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCBUN

```
COMCBUN   BUN2     SB7    B7+60D                                                  COMCBUN     151    A
COMCBUN            SA5    A5+1                                                    COMCBUN     152    A
COMCBUN            GE     B7,B0,BUN5  IF NO MORE SOURCE WORDS TO BU/BU            COMCBUN     153    A
COMCBUN            SB6    10*CHAR                                                 COMCBUN     154    A
COMCBUN                                                                           COMCBUN     155    A
COMCBUN   BUN3     LX5    CHAR                                                    COMCBUN     156    A
COMCBUN            LE     B6,B0,BUN2  IF SOURCE WORD (X5) IS EXHAUSTED            COMCBUN     157    A
COMCBUN            BX4    -X0*X5                                                  COMCBUN     158    A
COMCBUN                                                                           COMCBUN     159    A
COMCBUN   BUN4     SB6    B6-CHAR                                                 COMCBUN     160    A
COMCBUN            SB2    X4+10000B-BICH                                          COMCBUN     161    A
COMCBUN            LX7    X1,B2                                                   COMCBUN     162    A
COMCBUN            EQ     B5,B0,BUN6  IF CHARACTER COUNT EXHAUSTED                COMCBUN     163    A
COMCBUN            LX3    X4,B3                                                   COMCBUN     164    A
COMCBUN            PL     X7,BUN6     IF NEXT CHAR TO PACK NOT IN THIS GROUP      COMCBUN     165    A
COMCBUN            SB3    B3-CHAR                                                 COMCBUN     166    A
COMCBUN            SB5    B5-B1                                                   COMCBUN     167    A
COMCBUN            BX6    X6+X3                                                   COMCBUN     168    A
COMCBUN            GE     B3,B4,BUN3  IF PACKING REGISTER (X6) NOT FULL           COMCBUN     169    A
COMCBUN                                                                           COMCBUN     170    A
COMCBUN   *        HERE IF PACKING REGISTER (X6) IS FULL.                         COMCBUN     171    A
COMCBUN                                                                           COMCBUN     172    A
COMCBUN            MX3    -BN.TOTL                                                COMCBUN     173    A
COMCBUN            SA6    A6+B1                                                   COMCBUN     174    A
COMCBUN            SB3    BN.TOCP+BN.TOCL-CHAR                                    COMCBUN     175    A
COMCBUN            LX3    BN.TOTP-0                                               COMCBUN     176    A
COMCBUN            BX6    -X3*X6      CLEAR *TOC*                                 COMCBUN     177    A
COMCBUN            EQ     BUN3        BACK INTO THE FIRE...                       COMCBUN     178    A
COMCBUN                                                                           COMCBUN     179    A
COMCBUN   *        HERE AT FINAL/LAST SOURCE WORD.                                COMCBUN     180    A
COMCBUN   *                                                                       COMCBUN     181    A
COMCBUN   *          1. IF NEED TO BURST/BUILD FINAL SOURCE WORD, THEN GO BACK    COMCBUN     182    A
COMCBUN   *             AND DO IT.                                                COMCBUN     183    A
COMCBUN   *                                                                       COMCBUN     184    A
COMCBUN   *          2. IF HAVE ALREADY BU/BU FINAL SOURCE WORD, THEN DONE...     COMCBUN     185    A
COMCBUN                                                                           COMCBUN     186    A
COMCBUN    BUN5    SB6    A3-B7                                                   COMCBUN     187    A
COMCBUN            GT     B6,B0,BUN3  IF NEED TO BU/BU FINAL SOURCE WORD          COMCBUN     188    A
COMCBUN                                                                           COMCBUN     189    A
COMCBUN   *        FINAL PROCESSING...(X6) MIGHT NEED TO BE STORED OUT.           COMCBUN     190    A
COMCBUN                                                                           COMCBUN     191    A
COMCBUN    BUN6    SB4    B3-BN.TOCP-BN.TOCL+CHAR                                 COMCBUN     192    A
COMCBUN            SB7    A3-B7       RESTORE (B7) TO A CIVIL/HUMAN NUMBER        COMCBUN     193    A
COMCBUN            ZR     B4,EXIT.    IF (X6) IS EMPTY, DONE...                   COMCBUN     194    A
COMCBUN            SA6    A6+1                                                    COMCBUN     195    A
COMCBUN            EQ     EXIT.                                                   COMCBUN     196    A
COMCBUN    BUN     SPACE  4,10                                                    COMCBUN     197    A
COMCBUN            BASE   *                                                       COMCBUN     198    A
COMCBUN    QUAL$   IF     -DEF,QUAL$                                              COMCBUN     199    A
COMCBUN            QUAL   *                                                       COMCBUN     200    A
COMCBUN    BUN     =      /COMCBUN/BUN                                            COMCBUN     201    A
COMCBUN    QUAL$   ENDIF                                                          COMCBUN     202    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCBUN

COMCBUN   COMCBUN  ENDX                                                    COMCBUN      203     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - COMCBUN

          IDENTIFIER                TOTAL    ACTIVE

          COMCBUN                     203       203

          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

COMCIDP  *COMDECK COMCIDP                                                 COMCIDP        1     A
COMCIDP   COMCIDP  CTEXT  COMCIDP "IDPVER" - INTERACTIVE DEBUG PACKAGE.   COMCIDP        2     A
COMCIDP   COMCIDP  SPACE  4,10                                            COMCIDP        3     A
COMCIDP            IF     -DEF,QUAL$,1                                    COMCIDP        4     A
COMCIDP            QUAL   COMCIDP                                         COMCIDP        5     A
COMCIDP            BASE   D                                               COMCIDP        6     A
COMCIDP                                                                   COMCIDP        7     A
COMCIDP   IDP      MICRO  1,,/"QUAL"/                                     COMCIDP        8     A
COMCIDP   COMCIDP  SPACE  4,10                                            COMCIDP        9     A
COMCIDP  ***       COMCIDP - INTERACTIVE AND BATCH DEBUG PACKAGES.        COMCIDP       10     A
COMCIDP  *                                                                COMCIDP       11     A
COMCIDP  *                                                                COMCIDP       12     A
COMCIDP  *         CONTAINS THE INTERACTIVE DEBUG PACKAGE (IDP=), WHICH PLACES  COMCIDP  13  A
COMCIDP  *         A HOST PROGRAM IN INTERACTIVE MODE;  AND THE 2 BATCH   COMCIDP       14     A
COMCIDP  *         DEBUGGING ROUTINES (REG=,SNP=), WHICH PROVIDE REGISTER AND  COMCIDP  15  A
COMCIDP  *         CORE SNAPSHOT DUMPS.                                   COMCIDP       16     A
COMCIDP  *                                                                COMCIDP       17     A
COMCIDP  *         IN THE SECTIONS THAT FOLLOW, *IDP DEPENDENCIES* ARE DEFINED  COMCIDP  18  A
COMCIDP  *         TO BE THE INTERFACES THAT THE IDP INSTALLER MUST PROVIDE;  COMCIDP  19  A
COMCIDP  *         AND *IDP OPTIONS* ARE THOSE INTERFACES WHICH ARE OPTIONAL.  COMCIDP  20  A
COMCIDP  *         THAT IS, INTERFACES THAT HAVE DEFAULT CONDITIONS.      COMCIDP       21     A
COMCIDP  *                                                                COMCIDP       22     A
COMCIDP  *         IN THE CASE WHERE THE USER PROVIDES A SUBROUTINE THAT WILL  COMCIDP  23  A
COMCIDP  *         BE CALLED BY IDP, THE *ENTRY* AND *EXIT* CONDITIONS ARE SHOWN.  COMCIDP  24  A
COMCIDP  *         IN ADDITION, THE REGISTERS THAT CANNOT BE ALTERED BY THE USER  COMCIDP  25  A
COMCIDP  *         SUBROUTINE ARE SPECIFIED VIA *LOCKED*.  (B1) ARE ASSUMED TO BE  COMCIDP  26  A
COMCIDP  *         EQUAL TO 1 (LOCKED), AND WILL THEREFORE NOT BE MENTIONED  COMCIDP  27  A
COMCIDP  *         AGAIN.                                                 COMCIDP       28     A
COMCIDP  *                                                                COMCIDP       29     A
COMCIDP  ************************************************************** COMCIDP       30     A
COMCIDP  *                                                                COMCIDP       31     A
COMCIDP  *         IDP DEPENDENCIES--                                     COMCIDP       32     A
COMCIDP  *                                                                COMCIDP       33     A
COMCIDP  ************************************************************** COMCIDP       34     A
COMCIDP  *                                                                COMCIDP       35     A
COMCIDP  *         COMDECKS NEEDED BY *COMCIDP*.  ALTHOUGH THESE COMDECKS  COMCIDP  36  A
COMCIDP  *         ARE NOT ALL STRICTLY ASSEMBLY-TIME DEPENDENCIES (I.E. SOME CAN  COMCIDP  37  A
COMCIDP  *         BE SATISFIED AT LOAD-TIME), THE POINT IS THAT THEY ALL HAVE  COMCIDP  38  A
COMCIDP  *         TO BE AROUND FOR *COMCIDP* TO WORK.  SOME ARE THE CLASSIC  COMCIDP  39  A
COMCIDP  *         *KRONOS* COMDECKS AND SOME ARE NEW.  ALL CAN BE FOUND ON  COMCIDP  40  A
COMCIDP  *         *IDPPL* (IDP OLDPL), *OLDPL,ID=COMPASS*, OR *CPUTEXT*.  COMCIDP  41  A
COMCIDP  *                                                                COMCIDP       42     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *        WARNING-- IT IS SUGGESTED THAT THE IDP INSTALLER PROVIDE          COMCIDP     43    A
COMCIDP  *        SEPARATE COPIES OF CODE COMDECKS FOR *COMCIDP*, EVEN IF           COMCIDP     44    A
COMCIDP  *        THE HOST ALREADY HAS COPIES OF THESE COMDECKS.  THIS WILL         COMCIDP     45    A
COMCIDP  *        AVOID PROBLEMS THAT CAN OCCUR IN *STEP* MODE WHEN IDP             COMCIDP     46    A
COMCIDP  *        TRIES TO STEP COMDECKS THAT IT IS ALSO USING ITSELF.              COMCIDP     47    A
COMCIDP  *                                                                          COMCIDP     48    A
COMCIDP  *        ACTCOM    COMACIO   COMADEF   COMAFET   COMAIDP   COMAREG         COMCIDP     49    A
COMCIDP  *        COMATOK                                                           COMCIDP     50    A
COMCIDP  *                                                                          COMCIDP     51    A
COMCIDP  *        COMCBUB   COMCBUN   COMCCDD   COMCCIO   COMCCOD   COMCLFM         COMCIDP     52    A
COMCIDP  *        COMCMCS   COMCRDC   COMCRDW   COMCRSR   COMCSBM   COMCSFN         COMCIDP     53    A
COMCIDP  *        COMCSVR   COMCSYS   COMCTOK   COMCWOD   COMCWTC   COMCWTW         COMCIDP     54    A
COMCIDP  *        COMCXJR   COMCZTB                                                 COMCIDP     55    A
COMCIDP  *                                                                          COMCIDP     56    A
COMCIDP  *        COMSIDP   COMSRAS   COMSTOK                                       COMCIDP     57    A
COMCIDP  ************************************************************************  COMCIDP     58    A
COMCIDP  *                                                                          COMCIDP     59    A
COMCIDP  *        IDP OPTIONS--                                                     COMCIDP     60    A
COMCIDP  *                                                                          COMCIDP     61    A
COMCIDP  ************************************************************************  COMCIDP     62    A
COMCIDP  *                                                                          COMCIDP     63    A
COMCIDP  *        CP.NFLS - NOMINAL FIELD LENGTH SCM.                               COMCIDP     64    A
COMCIDP  *                                                                          COMCIDP     65    A
COMCIDP  *        IF THE SYMBOL *CP.NFLS* IS AVAILABLE (=Y), THEN IT IS THE         COMCIDP     66    A
COMCIDP  *        ADDR OF THE SCM LOCATION THAT CONTAINS THE CURRENT NOMINAL        COMCIDP     67    A
COMCIDP  *        FIELD LENGTH SCM FOR THE HOST PROGRAM.  THIS OPTION IS            COMCIDP     68    A
COMCIDP  *        INTENDED FOR HOST PROGRAMS THAT PERFORM THEIR OWN MEMORY          COMCIDP     69    A
COMCIDP  *        MANAGEMENT.  IF *CP.NFLS* IS AVAILABLE, THEN *IDP* USES ITS       COMCIDP     70    A
COMCIDP  *        CONTENTS FOR ADDR LEGALITY CHECKS AND ASSUMES THAT THE HOST       COMCIDP     71    A
COMCIDP  *        IS KEEPING IT UPDATED.                                            COMCIDP     72    A
COMCIDP  *                                                                          COMCIDP     73    A
COMCIDP  *              0 .LE. LEGAL ADDR .LT. (CP.NFLS)                            COMCIDP     74    A
COMCIDP  *                                                                          COMCIDP     75    A
COMCIDP  *        FORMAT --                                                         COMCIDP     76    A
COMCIDP  *              VFD   42/0,18/FL                                            COMCIDP     77    A
COMCIDP  *                                                                          COMCIDP     78    A
COMCIDP  *        IF THE SYMBOL *CP.NFLS* IS NOT AVAILABLE, THEN *IDP*              COMCIDP     79    A
COMCIDP  *        WILL CHECK SCM ADDRESSES USING THE FOLLOWING SCHEME--             COMCIDP     80    A
COMCIDP  *                                                                          COMCIDP     81    A
COMCIDP  *          1. IF BITS 0 THRU 17 OF (RA.LWP) ARE .MI., THEN *CMM*           COMCIDP     82    A
COMCIDP  *             IS ACTIVE.  *IDP* WILL THEREFORE TRY AND FIND THE            COMCIDP     83    A
COMCIDP  *             CURRENT FL AT (DABA), I.E. (-(RA.LWP)).  HOWEVER,            COMCIDP     84    A
COMCIDP  *             BECAUSE *CMM* IS NOT REAL GOOD AT KEEPING *DABA*             COMCIDP     85    A
COMCIDP  *             UPDATED PROPERLY, *IDP* WILL DOUBLE CHECK AN                 COMCIDP     86    A
COMCIDP  *             ADDR THAT *DABA* INDICATES IS OUT-OF-RANGE BY PERFORMING     COMCIDP     87    A
COMCIDP  *             A SYSTEM *MEMORY* (*MEM* REQUEST) TO FIND OUT THE TRUE       COMCIDP     88    A
COMCIDP  *             FL.  NOTE-- *IDP* DOES NOT UNCONDITIONALLY CHECK ADDRS       COMCIDP     89    A
COMCIDP  *             AGAINST A *MEMORY* FOR EFFICIENCY REASONS.                   COMCIDP     90    A
COMCIDP  *                                                                          COMCIDP     91    A
COMCIDP  *          2. IF BITS 0 THRU 17 OF (RA.LWP) ARE .PL., THEN ADDRESSES       COMCIDP     92    A
COMCIDP  *             ARE CHECKED AGAINST A *MEMORY* REQUEST.                      COMCIDP     93    A
COMCIDP  *                                                                          COMCIDP     94    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  ***********************************************************************  COMCIDP     95    A
COMCIDP  *                                                                        COMCIDP     96    A
COMCIDP  *         EOS - END-OF-STATEMENT MICRO.                                  COMCIDP     97    A
COMCIDP  *                                                                        COMCIDP     98    A
COMCIDP  *         IF THE SYMBOL *EOS* IS A MICRO NAME (MIC), THEN                COMCIDP     99    A
COMCIDP  *         "EOS" IS A MICRO WHICH DEFINES THE INTERACTIVE                 COMCIDP    100    A
COMCIDP  *         END-OF-STATEMENT CHARACTER.  THE USE OF THIS CHARACTER ALLOWS  COMCIDP    101    A
COMCIDP  *         AN *IDP* USER TO ENTER MORE THAN ONE INTERACTIVE COMMAND ON    COMCIDP    102    A
COMCIDP  *         A SINGLE LINE.                                                 COMCIDP    103    A
COMCIDP  *                                                                        COMCIDP    104    A
COMCIDP  *         DEFAULT IS --                                                  COMCIDP    105    A
COMCIDP  *                                                                        COMCIDP    106    A
COMCIDP  *         EOS    MICRO  1,,/;/                                           COMCIDP    107    A
COMCIDP  *                                                                        COMCIDP    108    A
COMCIDP  *         E.G.   SNAP,100;REG,X1;END                                     COMCIDP    109    A
COMCIDP  *                                                                        COMCIDP    110    A
COMCIDP  *         NOTE   THE END OF STATEMENT LOGIC WILL NOT WORK PROPERLY IF    COMCIDP    111    A
COMCIDP  *                "EOS" IS ONE OF THE FOLLOWING 8 CHARACTERS --           COMCIDP    112    A
COMCIDP  *                JKLMNOPQ                                                COMCIDP    113    A
COMCIDP  ***********************************************************************  COMCIDP    114    A
COMCIDP  *                                                                        COMCIDP    115    A
COMCIDP  *         FAA= - FIND ABSOLUTE ADDRESS.                                  COMCIDP    116    A
COMCIDP  *                                                                        COMCIDP    117    A
COMCIDP  *         IF THE SYMBOL *FAA=* IS AVAILABLE (=Y), THEN *FAA= - FIND      COMCIDP    118    A
COMCIDP  *         ABSOLUTE ADDRESS* IS A USER SUBROUTINE THAT WILL PROVIDE       COMCIDP    119    A
COMCIDP  *         *COMCIDP* WITH AN ABSOLUTE ADDRESS ASSOCIATED WITH A *NAME*.   COMCIDP    120    A
COMCIDP  *         THIS OPTION IS INTENDED FOR HOST PROGRAMS THAT CONSIST OF      COMCIDP    121    A
COMCIDP  *         MULTIPLE *DECK*S OR *IDENT*S, AND WHERE USERS OF *IDP* WOULD   COMCIDP    122    A
COMCIDP  *         LIKE TO BE ABLE TO REFERENCE ADDRESSES RELATIVE TO A           COMCIDP    123    A
COMCIDP  *         DECK NAME--  E.G. SNAP NAME+20,,10                             COMCIDP    124    A
COMCIDP  *                                                                        COMCIDP    125    A
COMCIDP  *         IF *FAA=* IS AVAILABLE, THEN *IDP* WILL *RJ =XFAA=* TO TRY TO  COMCIDP    126    A
COMCIDP  *         ASSOCIATE AN ABSOLUTE ADDRESS WITH *NAME* BEFORE HE SEARCHS    COMCIDP    127    A
COMCIDP  *         THE USER *SET* TABLE.                                          COMCIDP    128    A
COMCIDP  *                                                                        COMCIDP    129    A
COMCIDP  **        FAA= - FIND ABSOLUTE ADDRESS.                                  COMCIDP    130    A
COMCIDP  *                                                                        COMCIDP    131    A
COMCIDP  *         ENTRY  (X0) = MX0 7*CHAR     (CHAR=6)                          COMCIDP    132    A
COMCIDP  *                (X1) = 42/0LNAME, 18/0                                  COMCIDP    133    A
COMCIDP  *                                                                        COMCIDP    134    A
COMCIDP  *         EXIT   (X6) = .PL. IF ABS ADDR ASSOCIATED WITH NAME            COMCIDP    135    A
COMCIDP  *                       .MI. IF NO ADDR KNOWN FOR *NAME* (I.E. NO FIND)  COMCIDP    136    A
COMCIDP  *                                                                        COMCIDP    137    A
COMCIDP  *         LOCKED X - 0,1                                                 COMCIDP    138    A
COMCIDP  ***********************************************************************  COMCIDP    139    A
COMCIDP  *                                                                        COMCIDP    140    A
COMCIDP  *         FRA= - FIND RELATIVE ADDRESS.                                  COMCIDP    141    A
COMCIDP  *                                                                        COMCIDP    142    A
COMCIDP  *         IF THE SYMBOL *FRA=* IS AVAILABLE (=Y), THEN *FRA= - FIND      COMCIDP    143    A
COMCIDP  *         RELATIVE ADDRESS* IS A USER SUBROUTINE THAT WILL PROVIDE       COMCIDP    144    A
COMCIDP  *         *COMCIDP* WITH A DISPLAY CODE (DPC) NAME AND RELATIVE OFFSET   COMCIDP    145    A
COMCIDP  *         ASSOCIATED WITH AN ABSOLUTE ADDRESS. THIS ROUTINE IS THE       COMCIDP    146    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *        LOGICAL COMPLEMENT TO *FAA=*, AND IS USED BY ROUTINES IN          COMCIDP    147    A
COMCIDP  *        *COMCIDP* TO OUTPUT A MORE HUMAN READABLE ADDRESS FORMAT.         COMCIDP    148    A
COMCIDP  *                                                                          COMCIDP    149    A
COMCIDP  *        E.G. IN RESPONSE TO *IDP* COMMAND-- CODE NAME+10                  COMCIDP    150    A
COMCIDP  *                                                                          COMCIDP    151    A
COMCIDP  *        1025   010001023         RJ    1023     6 IN NAME                 COMCIDP    152    A
COMCIDP  *                                                                          COMCIDP    153    A
COMCIDP  *        IF *FRA=* IS AVAILABLE, WHENEVER A ROUTINE IN *COMCIDP*           COMCIDP    154    A
COMCIDP  *        REQUIRES THIS RELATIVE ADDRESS PLUS OFFSET DPC FORMAT, A          COMCIDP    155    A
COMCIDP  *        *EQ =XFRA=* IS EXECUTED.                                          COMCIDP    156    A
COMCIDP  *                                                                          COMCIDP    157    A
COMCIDP  *        IT IS SUGGESTED THAT THE IDP INSTALLER USE THE COMDECK            COMCIDP    158    A
COMCIDP  *        NAMED *CCOMRPV* (COMPILER COMMON REPRIEVE PROCESSOR).             COMCIDP    159    A
COMCIDP  *        *CCOMRPV* CONTAINS, AMONG OTHER THINGS, A REASONABLE VERSION      COMCIDP    160    A
COMCIDP  *        OF *FRA=*.  SEE *CCOMRPV*.                                        COMCIDP    161    A
COMCIDP  *                                                                          COMCIDP    162    A
COMCIDP  *        SEE ALSO PREAMBLE TO SUBROUTINE *FRA* (FIND RELATIVE ADDRESS)     COMCIDP    163    A
COMCIDP  *        IN *COMCIDP*.                                                     COMCIDP    164    A
COMCIDP  *                                                                          COMCIDP    165    A
COMCIDP  **       FRA= - FIND RELATIVE ADDRESS.                                     COMCIDP    166    A
COMCIDP  *                                                                          COMCIDP    167    A
COMCIDP  *        ENTRY  (X1) = 60/ABS ADDR                                         COMCIDP    168    A
COMCIDP  *               (B7) = RETURN ADDRESS-- AN *RJ =XFRA=* IS NOT USED         COMCIDP    169    A
COMCIDP  *                      BECAUSE IF THE HOST PROGRAM HAS REPRIEVE            COMCIDP    170    A
COMCIDP  *                      PROCESSING, THEN *FRA=* COULD BE INCORPORATED       COMCIDP    171    A
COMCIDP  *                      INTO IT AND EXIST IN AN AREA CHECKSUMMED            COMCIDP    172    A
COMCIDP  *                      BY *RPV*                                            COMCIDP    173    A
COMCIDP  *                                                                          COMCIDP    174    A
COMCIDP  *        EXIT   (X6) = 1ST WORD OF DPC RESULT                              COMCIDP    175    A
COMCIDP  *                      E.G. (X6) = NNNNNN.IN.     (.=BLANK(55B))           COMCIDP    176    A
COMCIDP  *               (X7) = 2ND WORD OF DPC RESULT  (-C- FORMAT)                COMCIDP    177    A
COMCIDP  *                      E.G. (X7) = XXXXXXX000     (0=00B)                  COMCIDP    178    A
COMCIDP  *                                                                          COMCIDP    179    A
COMCIDP  *        LOCKED X - 5                                                      COMCIDP    180    A
COMCIDP  *               A - 0,5,6,7                                                COMCIDP    181    A
COMCIDP  ***********************************************************************   COMCIDP    182    A
COMCIDP  *                                                                          COMCIDP    183    A
COMCIDP  *        PROMPT - INTERACTIVE PROMPT MICRO NAME                            COMCIDP    184    A
COMCIDP  *                                                                          COMCIDP    185    A
COMCIDP  *        IF THE SYMBOL *PROMPT* IS A MICRO NAME (MIC), THEN                COMCIDP    186    A
COMCIDP  *        "PROMPT" IS A MICRO WHICH DEFINES THE INTERACTIVE PROMPT.         COMCIDP    187    A
COMCIDP  *        THIS PROMPT WILL BE ISSUED WHENEVER *IDP* REQUIRES THAT THE       COMCIDP    188    A
COMCIDP  *        USER ENTER A COMMAND.                                             COMCIDP    189    A
COMCIDP  *                                                                          COMCIDP    190    A
COMCIDP  *        DEFAULT IS --                                                     COMCIDP    191    A
COMCIDP  *                                                                          COMCIDP    192    A
COMCIDP  *        PROMPT MICRO  1,,/>>/                                             COMCIDP    193    A
COMCIDP  *                                                                          COMCIDP    194    A
COMCIDP  *        NOTE - *IDP* PREFIXES THE "PROMPT" WITH A BLANK (55B) PRINT       COMCIDP    195    A
COMCIDP  *               CONTROL CHARACTER.                                         COMCIDP    196    A
COMCIDP  ***********************************************************************   COMCIDP    197    A
COMCIDP  *                                                                          COMCIDP    198    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *        UKT=/FW=UKT - USER KEYWORD TABLE DEFINITION AND LINKING.          COMCIDP    199    A
COMCIDP  *                                                                          COMCIDP    200    A
COMCIDP  *        IF THE SYMBOL *UKT=* IS AVAILABLE (=Y), THEN *UKT=* IS THE        COMCIDP    201    A
COMCIDP  *        FWA OF THE USER KEYWORD TABLE.  THIS OPTION ALLOWS THE USER TO    COMCIDP    202    A
COMCIDP  *        PERFORM HER OR HIS OWN KEYWORD PROCESSING IN INTERACTIVE MODE,    COMCIDP    203    A
COMCIDP  *        THEREBY USING IDP AS A TERMINAL HANDLER FOR ONES OWN SCHEMES.     COMCIDP    204    A
COMCIDP  *                                                                          COMCIDP    205    A
COMCIDP  *        IN GENERAL TERMS, THIS WORKS AS FOLLOWS: AFTER IDP HAS            COMCIDP    206    A
COMCIDP  *        SEARCHED ITS OWN KEYWORD TABLE, HE WILL SEARCH THE USER           COMCIDP    207    A
COMCIDP  *        KEYWORD TABLE AT *UKT=* ET SEQ.  IF IDP FINDS A KEYWORD MATCH     COMCIDP    208    A
COMCIDP  *        IN THE USER'S KEYWORD TABLE, THEN IDP TRANSFERS CONTROL TO        COMCIDP    209    A
COMCIDP  *        A SPECIFIED USER PROCESSOR.                                       COMCIDP    210    A
COMCIDP  *                                                                          COMCIDP    211    A
COMCIDP  *        NOTE THAT AS STATED ABOVE, THIS ALGORITHM GIVES PRECEDENCE        COMCIDP    212    A
COMCIDP  *        TO IDP'S KEYWORDS OVER THE USER'S KEYWORDS.                       COMCIDP    213    A
COMCIDP  *                                                                          COMCIDP    214    A
COMCIDP  *        THE KEYWORD TABLE(S) FOR IDP CAN BE LINKED TOGETHER IN THE        COMCIDP    215    A
COMCIDP  *        FOLLOWING WAY --                                                  COMCIDP    216    A
COMCIDP  *                                                                          COMCIDP    217    A
COMCIDP  *              ++++++++++++++++++++                                        COMCIDP    218    A
COMCIDP  *              +                  +                                        COMCIDP    219    A
COMCIDP  *              + IDP KEYWORD TABLE +                                       COMCIDP    220    A
COMCIDP  *              +                  +                                        COMCIDP    221    A
COMCIDP  *              ++++++++++++++++++++  .....                                 COMCIDP    222    A
COMCIDP  *                                         .                                COMCIDP    223    A
COMCIDP  *              ++++++++++++++++++++  <....                                 COMCIDP    224    A
COMCIDP  *              +                  +                                        COMCIDP    225    A
COMCIDP  *              + UKT NUMBER 1     +                                        COMCIDP    226    A
COMCIDP  *              +                  +                                        COMCIDP    227    A
COMCIDP  *              ++++++++++++++++++++  .....                                 COMCIDP    228    A
COMCIDP  *                                         .                                COMCIDP    229    A
COMCIDP  *                      .                  .                                COMCIDP    230    A
COMCIDP  *                      .                  .                                COMCIDP    231    A
COMCIDP  *              ++++++++++++++++++++  <....                                 COMCIDP    232    A
COMCIDP  *              +                  +                                        COMCIDP    233    A
COMCIDP  *              + UKT NUMBER N     +                                        COMCIDP    234    A
COMCIDP  *              +                  +                                        COMCIDP    235    A
COMCIDP  *              ++++++++++++++++++++                                        COMCIDP    236    A
COMCIDP  *                                                                          COMCIDP    237    A
COMCIDP  *        THIS IMPLIES, THEREFORE, THAT THERE CAN BE ANY NUMBER OF USER     COMCIDP    238    A
COMCIDP  *        KEYWORD TABLES LINKED TO IDP AT ANY TIME.                         COMCIDP    239    A
COMCIDP  *                                                                          COMCIDP    240    A
COMCIDP  *        EACH KEYWORD TABLE IS TERMINATED BY A FULL ZERO WORD.  A 2ND      COMCIDP    241    A
COMCIDP  *        FULL WORD IMMEDIATELY FOLLOWING THE END-OF-TABLE WORD IS USED     COMCIDP    242    A
COMCIDP  *        FOR LINKING TO MORE KEYWORD TABLES, IF DESIRED.  IT SHOULD        COMCIDP    243    A
COMCIDP  *        CONTAIN THE FWA OF THE **NEXT** USER KEYWORD TABLE, OR BE         COMCIDP    244    A
COMCIDP  *        ZERO TO INDICATE END-OF-ALL-KEYWORD-TABLES.                       COMCIDP    245    A
COMCIDP  *                                                                          COMCIDP    246    A
COMCIDP  *        EACH INDIVIDUAL KEYWORD TABLE ENTRY SHOULD BE DEFINED VIA THE     COMCIDP    247    A
COMCIDP  *        *KEYW* MACRO IN *COMSIDP* (IDP INTERFACE TEXT).                   COMCIDP    248    A
COMCIDP  *                                                                          COMCIDP    249    A
COMCIDP  *        AFTER KEYWORD PROCESSING, THE USER SHOULD RETURN CONTROL BACK     COMCIDP    250    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *         TO IDP BY EXECUTING A BRANCH TO *IDP=MN* (IDP MAIN LOOP NODE)          COMCIDP      251     A
COMCIDP  *         IF THERE WERE NO ERRORS, OR TO ONE OF IDP'S ERROR PROCESSORS:          COMCIDP      252     A
COMCIDP  *                                                                                COMCIDP      253     A
COMCIDP  *         *IDP=ER* IS INVOKED FOR USER DETECTED ERRORS FOR WHICH THERE           COMCIDP      254     A
COMCIDP  *         IS NO ERROR MESSAGE.  THIS PROCESSOR EXISTS PRIMARILY FOR              COMCIDP      255     A
COMCIDP  *         COMPATABILITY WITH AN OLDER VERSION OF IDP CALLED *DBG=IDP*.           COMCIDP      256     A
COMCIDP  *         I ADVISE **NOT** USING THIS ERROR PROCESSOR FOR THE SIMPLE             COMCIDP      257     A
COMCIDP  *         REASON THAT ERROR MESSAGES ARE NICE THINGS TO HAVE...                  COMCIDP      258     A
COMCIDP  *                                                                                COMCIDP      259     A
COMCIDP  *         THE MACRO *ERROR* (DEFINED IN *COMSIDP*) IS USED TO SPECIFY            COMCIDP      260     A
COMCIDP  *         AN ERROR THAT **DOES** HAVE AN ERROR MESSAGE.  SEE *COMSIDP*.          COMCIDP      261     A
COMCIDP  *                                                                                COMCIDP      262     A
COMCIDP  *         THE MACRO *SYSERR* (DEFINED IN *COMSIDP*) IS USED FOR ERRORS           COMCIDP      263     A
COMCIDP  *         THAT ARE DETECTED VIA *SELF CHECKING* TYPE CODE.  THESE ARE            COMCIDP      264     A
COMCIDP  *         INTERNAL TYPE IDP SYSTEM ERRORS THAT COULD POTENTIALLY BE              COMCIDP      265     A
COMCIDP  *         VERY DANGEROUS/CONFUSING TO THE DEBUGGER WHO IS NOT INTERESTED         COMCIDP      266     A
COMCIDP  *         IN DEBUGGING IDP ITSELF.  IDP CONTAINS A GOODLY AMOUNT OF              COMCIDP      267     A
COMCIDP  *         SELF-CHECKING CODE THAT AIMS AT MINIMIZING IDP SCREW-UPS.              COMCIDP      268     A
COMCIDP  *                                                                                COMCIDP      269     A
COMCIDP  ************************************************************************        COMCIDP      270     A
COMCIDP  *                                                                                COMCIDP      271     A
COMCIDP  *         UIO= - USER IDP OWNCODE.                                               COMCIDP      272     A
COMCIDP  *                                                                                COMCIDP      273     A
COMCIDP  *         IF THE SYMBOL *UIO=* IS AVAILABLE (=Y), THEN *UIO= - USER              COMCIDP      274     A
COMCIDP  *         IDP OWNCODE* IS A USER SUBROUTINE THAT IS CALLED AFTER ENTRY           COMCIDP      275     A
COMCIDP  *         TO *IDP=*. IT ALLOWS THE USER TO PERFORM HER OWN TASKS BEFORE          COMCIDP      276     A
COMCIDP  *         ENTERING INTERACTIVE MODE (E.G. CHECKING A MASTER SNAP FLAG).          COMCIDP      277     A
COMCIDP  *                                                                                COMCIDP      278     A
COMCIDP  **        UIO= - USER IDP OWNCODE.                                               COMCIDP      279     A
COMCIDP  *                                                                                COMCIDP      280     A
COMCIDP  *         ENTRY  NONE                                                            COMCIDP      281     A
COMCIDP  *                                                                                COMCIDP      282     A
COMCIDP  *         EXIT   (X1) = .MI. IF TO ENTER INTERACTIVE MODE                        COMCIDP      283     A
COMCIDP  *                     = .PL. IF NOT ENTERING INTERACTIVE MODE THIS TIME          COMCIDP      284     A
COMCIDP  *                            I.E. ALL REGISTERS ARE RESTORED AND                 COMCIDP      285     A
COMCIDP  *                                 CONTROL RETURNS TO CALLER                      COMCIDP      286     A
COMCIDP  *                                                                                COMCIDP      287     A
COMCIDP  *         LOCKED X - 5                                                           COMCIDP      288     A
COMCIDP  ************************************************************************        COMCIDP      289     A
COMCIDP  *                                                                                COMCIDP      290     A
COMCIDP  *         .OS - DEFINE OPERATING SYSTEM.                                         COMCIDP      291     A
COMCIDP  *                                                                                COMCIDP      292     A
COMCIDP  *         .OS = 1   KRONOS 2 OR NOS 1/TS                                         COMCIDP      293     A
COMCIDP  *         .OS = 2   SCOPE 2  (CYBER 76)                                          COMCIDP      294     A
COMCIDP  *         .OS = 3   SCOPE 3 OR NOS 1/BE  (DEFAULT)                               COMCIDP      295     A
COMCIDP  ************************************************************************        COMCIDP      296     A
COMCIDP  *                                                                                COMCIDP      297     A
COMCIDP  *         URO= - USER *REG=* OWNCODE.                                            COMCIDP      298     A
COMCIDP  *                                                                                COMCIDP      299     A
COMCIDP  *         IF THE SYMBOL *URO=* IS AVAILABLE (=Y), THEN *URO= - USER              COMCIDP      300     A
COMCIDP  *         REG= OWNCODE* IS A USER SUBROUTINE THAT IS CALLED AFTER ENTRY          COMCIDP      301     A
COMCIDP  *         TO THE REGISTER SNAPSHOT ROUTINE *REG=* (CALLED VIA *REG*              COMCIDP      302     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *         MACRO IN *COMAIDP*). IT ALLOWS THE USER TO PERFORM HIS OWN          COMCIDP    303    A
COMCIDP  *         TASKS BEFORE THE REGISTER SNAPSHOT IS TAKEN (E.G. CHECKING A        COMCIDP    304    A
COMCIDP  *         MASTER SNAP FLAG).                                                  COMCIDP    305    A
COMCIDP  *                                                                             COMCIDP    306    A
COMCIDP  **        URO= - USER REG= OWNCODE.                                           COMCIDP    307    A
COMCIDP  *                                                                             COMCIDP    308    A
COMCIDP  *         ENTRY  (X5)+SN=URF = ADDRESS OF USER FLAGS                          COMCIDP    309    A
COMCIDP  *                                                                             COMCIDP    310    A
COMCIDP  *         EXIT   (X1) = .MI. IF TO PERFORM REGISTER SNAPSHOT                  COMCIDP    311    A
COMCIDP  *                     = .PL. IF NO REGISTER SNAPSHOT THIS TIME                COMCIDP    312    A
COMCIDP  ************************************************************************      COMCIDP    313    A
COMCIDP  *                                                                             COMCIDP    314    A
COMCIDP  *         USO= - USER *SNP=* OWNCODE.                                         COMCIDP    315    A
COMCIDP  *                                                                             COMCIDP    316    A
COMCIDP  *         IF THE SYMBOL *USO=* IS AVAILABLE (=Y), THEN *USO= - USER           COMCIDP    317    A
COMCIDP  *         SNP= OWNCODE* IS A USER SUBROUTINE THAT IS CALLED AFTER ENTRY       COMCIDP    318    A
COMCIDP  *         TO THE CORE SNAPSHOT ROUTINE *SNP=* (CALLED VIA *SNAP* MACRO        COMCIDP    319    A
COMCIDP  *         IN *COMAIDP*). IT ALLOWS THE USER TO PERFORM HIS OWN TASKS          COMCIDP    320    A
COMCIDP  *         BEFORE THE SNAPSHOT IS TAKEN (E.G. CHECKING A MASTER SNAP           COMCIDP    321    A
COMCIDP  *         FLAG).                                                              COMCIDP    322    A
COMCIDP  *                                                                             COMCIDP    323    A
COMCIDP  **        USO= - USER SNP= OWNCODE.                                           COMCIDP    324    A
COMCIDP  *                                                                             COMCIDP    325    A
COMCIDP  *         ENTRY  (X5)+SN=USF = ADDRESS OF USER FLAGS                          COMCIDP    326    A
COMCIDP  *                                                                             COMCIDP    327    A
COMCIDP  *         EXIT   (X1) = .MI. IF SNAP TO BE TAKEN                              COMCIDP    328    A
COMCIDP  *                     = .PL. IF NO SNAP THIS TIME                             COMCIDP    329    A
COMCIDP  ************************************************************************      COMCIDP    330    A
COMCIDP  *                                                                             COMCIDP    331    A
COMCIDP  *         USY= - USER SYMBOL TABLES SEARCH.                                   COMCIDP    332    A
COMCIDP  *                                                                             COMCIDP    333    A
COMCIDP  *         IF THE SYMBOL *USY=* IS AVAILABLE (=Y), THEN *USY=* (USER           COMCIDP    334    A
COMCIDP  *         SYMBOL TABLES SEARCH) IS A USER SUBROUTINE THAT IS CALLED           COMCIDP    335    A
COMCIDP  *         WHEN IDP WISHES TO ASSOCIATE A BINARY VALUE WITH A DPC              COMCIDP    336    A
COMCIDP  *         SYMBOL.                                                             COMCIDP    337    A
COMCIDP  *                                                                             COMCIDP    338    A
COMCIDP  *         FOR EXAMPLE, IF THE INTERACTIVE PERSON TYPES --                     COMCIDP    339    A
COMCIDP  *                                                                             COMCIDP    340    A
COMCIDP  *                SNAP FROM,TO                                                 COMCIDP    341    A
COMCIDP  *                                                                             COMCIDP    342    A
COMCIDP  *         AND IDP CAN NOT FIND THE SYMBOLS *FROM* AND/OR *TO* VIA             COMCIDP    343    A
COMCIDP  *         ITS OWN SEARCHING, THEN IDP WILL TRY AND CALL *USY=* TO             COMCIDP    344    A
COMCIDP  *         ALLOW THE HOST TO TRY AND ASSOCIATE A BINARY VALUE WITH             COMCIDP    345    A
COMCIDP  *         THIS SO FAR UNDEFINED SYMBOL.                                       COMCIDP    346    A
COMCIDP  *                                                                             COMCIDP    347    A
COMCIDP  **        USY= - USER SYMBOL TABLES SEARCH.                                   COMCIDP    348    A
COMCIDP  *                                                                             COMCIDP    349    A
COMCIDP  *         ENTRY  (X1) = SYMBOL TO SEARCH FOR, -L- FMT.                        COMCIDP    350    A
COMCIDP  *                                                                             COMCIDP    351    A
COMCIDP  *         EXIT   (X1) = .NZ. IF A FIND, ELSE .ZR.                             COMCIDP    352    A
COMCIDP  *                (X6) = BINARY VALUE TO ASSOCIATE WITH THIS SYMBOL.           COMCIDP    353    A
COMCIDP  *                                                                             COMCIDP    354    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *         LOCKED A0,X0,A5,X5                                    COMCIDP    355    A
COMCIDP  *                                                               COMCIDP    356    A
COMCIDP  ************************************************************   COMCIDP    357    A
COMCIDP   CODING   SPACE  4,10                                           COMCIDP    358    A
COMCIDP  ***       COMMENTING/CODING CONVENTIONS.                        COMCIDP    359    A
COMCIDP  *                                                               COMCIDP    360    A
COMCIDP  *                                                               COMCIDP    361    A
COMCIDP  *         A NUMBER OF COMMENTING AND CODING CONVENTIONS HAVE BEEN COMCIDP  362    A
COMCIDP  *         ADOPTED IN THIS PIECE OF CODE WHICH SHOULD MAKE IT EASIER COMCIDP  363    A
COMCIDP  *         TO READ AND UNDERSTAND.  THIS AUTHOR BELIEVES VERY STRONGLY COMCIDP  364    A
COMCIDP  *         THAT THE **NOTATION** USED TO DESCRIBE A PARTICULAR DESIGN COMCIDP  365    A
COMCIDP  *         IMPLEMENTATION IS AS IMPORTANT AS THE DESIGN ITSELF.  AND COMCIDP  366    A
COMCIDP  *         INSTEAD OF REQUIRING THAT YOU DECIPHER **MY** NOTATION ON COMCIDP  367    A
COMCIDP  *         THE FLY, I WILL TELL YOU HERE AND NOW HOW IT WORKS...   COMCIDP    368    A
COMCIDP  *                                                               COMCIDP    369    A
COMCIDP  *         THIS DISCUSSION CAN BE DIVIDED INTO 3 PARTS: CODE FORM, COMCIDP  370    A
COMCIDP  *         SYMBOL NAMING, AND A GLOSSARY OF ABBREVIATIONS.         COMCIDP    371    A
COMCIDP  *                                                               COMCIDP    372    A
COMCIDP  ************************************************************   COMCIDP    373    A
COMCIDP  *                                                               COMCIDP    374    A
COMCIDP  *         CODE FORM                                             COMCIDP    375    A
COMCIDP  *                                                               COMCIDP    376    A
COMCIDP  *         THIS IS A DESCRIPTION OF HOW ONE CAN EXPECT CODE TO LOOK COMCIDP  377    A
COMCIDP  *         ON A PAGE.  THE *MACE/KRONOS* CONVENTION IS THE MODEL.  COMCIDP    378    A
COMCIDP  *         *MACE* IS A VERY STYLIZED CODING CONVENTION WHICH IS AIMED COMCIDP  379    A
COMCIDP  *         PRIMARILY AT FORMALIZING SUBROUTINE STRUCTURE.          COMCIDP    380    A
COMCIDP  *                                                               COMCIDP    381    A
COMCIDP  *         *MACE*, BRIEFLY --                                    COMCIDP    382    A
COMCIDP  *                                                               COMCIDP    383    A
COMCIDP  *           1. SUBROUTINES ARE USED THAT ENTER AND EXIT AT A COMMON COMCIDP  384    A
COMCIDP  *              POINT, I.E. ARE INVOKED BY AN *RJ* INSTRUCTION.  COMCIDP    385    A
COMCIDP  *                                                               COMCIDP    386    A
COMCIDP  *           2. ALL SUBROUTINES HAVE 3 CHARACTER MNEMONIC NAMES  COMCIDP    387    A
COMCIDP  *              THAT DESCRIBE THEIR FUNCTION.                    COMCIDP    388    A
COMCIDP  *                                                               COMCIDP    389    A
COMCIDP  *           3. ALL SUBROUTINES BEGIN WITH AN EXTREMELY STYLIZED COMCIDP  390    A
COMCIDP  *              PREAMBLE WHICH GIVES THE NAME OF THE SUBROUTINE, ITS COMCIDP  391    A
COMCIDP  *              FUNCTION, ITS ENTRY AND EXIT CONDITIONS, THE REGISTERS COMCIDP  392    A
COMCIDP  *              IT USES/DESTROYS, AND A LIST OF ALL THE SUBROUTINES COMCIDP  393    A
COMCIDP  *              AND/OR MACROS THAT IT CALLS/REFERENCES.          COMCIDP    394    A
COMCIDP  *                                                               COMCIDP    395    A
COMCIDP  *              A DETAILED DESCRIPTION OF THE PREAMBLE IS BASICALLY COMCIDP  396    A
COMCIDP  *              A WASTE OF TIME.  THEY ARE ALL EXACTLY ALIKE...JUST GO COMCIDP  397    A
COMCIDP  *              LOOK AT ONE.                                     COMCIDP    398    A
COMCIDP  *                                                               COMCIDP    399    A
COMCIDP  *           4. ALL SUBROUTINE EXECUTABLE CODE LABELS CONSIST OF THE COMCIDP  400    A
COMCIDP  *              SUBROUTINE 3 CHARACTER MNEMONIC FOLLOWED BY A DIGIT COMCIDP  401    A
COMCIDP  *              OR DIGITS IN ASCENDING ORDER.                    COMCIDP    402    A
COMCIDP  *                                                               COMCIDP    403    A
COMCIDP  *           5. SUBROUTINE LOCAL-ONLY SCRATCH CELLS HAVE LABELS THAT COMCIDP  404    A
COMCIDP  *              CONSIST OF THE SUBROUTINE 3 CHARACTER MNEMONIC FOLLOWED COMCIDP  405    A
COMCIDP  *              BY A SINGLE LETTER (BEGINNING AT -A-) IN ASCENDING COMCIDP  406    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *              ORDER.                                                  COMCIDP     407    A
COMCIDP  *                                                                      COMCIDP     408    A
COMCIDP  *           6. THE *COMPASS* 2,11,18,30 TABBING CONVENTION IS STRICTLY COMCIDP     409    A
COMCIDP  *              FOLLOWED.  I.E. LABEL FIELD IN COL 2, OPERATION FIELD   COMCIDP     410    A
COMCIDP  *              IN COL 11, ADDRESS FIELD IN COL 18, AND COMMENT FIELD   COMCIDP     411    A
COMCIDP  *              IN COL 30.                                              COMCIDP     412    A
COMCIDP  *                                                                      COMCIDP     413    A
COMCIDP  *           7. COMMENTS THAT DESCRIBE HARDWARE FUNCTION ARE AVOIDED.   COMCIDP     414    A
COMCIDP  *              E.G.  SA1  ZIP    LOAD FLAG                             COMCIDP     415    A
COMCIDP  *                    LX1  4      LEFT-SHIFT (X1)                       COMCIDP     416    A
COMCIDP  *                                                                      COMCIDP     417    A
COMCIDP  *           8. **ALL** CONDITIONAL BRANCH INSTRUCTIONS CONTAIN A COMMENT COMCIDP     418    A
COMCIDP  *              IN COLS 30-72 THAT BEGINS WITH THE WORD *IF* AND        COMCIDP     419    A
COMCIDP  *              DESCRIBES THE *BRANCH TAKEN* CONDITION.  E.G.           COMCIDP     420    A
COMCIDP  *                                                                      COMCIDP     421    A
COMCIDP  *                    ZR   X1,EXIT. IF AT END OF TABLE                  COMCIDP     422    A
COMCIDP  *                                                                      COMCIDP     423    A
COMCIDP  *        FOR ME, *MACE*S PRIMARY VIRTUE IS THAT IS MAKES FOREIGN CODE  COMCIDP     424    A
COMCIDP  *        LOOK NOT-SO-FOREIGN.  SIMPLE.                                 COMCIDP     425    A
COMCIDP  *                                                                      COMCIDP     426    A
COMCIDP  *        IN ADDITION, THIS CODE USES A FEW EXTENSIONS TO THE *MACE*    COMCIDP     427    A
COMCIDP  *        WAY... THEY ARE --                                            COMCIDP     428    A
COMCIDP  *                                                                      COMCIDP     429    A
COMCIDP  *           1. THE COMMENT FIELD FOR A LINE CONTAINING A MACHINE       COMCIDP     430    A
COMCIDP  *              MNEMONIC INSTRUCTIONS (I.E. COLS 30-72) IS GENERALLY    COMCIDP     431    A
COMCIDP  *              RESERVED FOR A DESCRIPTION OF THE DATA STRUCTURE        COMCIDP     432    A
COMCIDP  *              RESULTING FROM THE HARDWARE INSTRUCTION ON THAT LINE.   COMCIDP     433    A
COMCIDP  *                                                                      COMCIDP     434    A
COMCIDP  *              USUALLY, THIS IS NOTATED VERY FORMALLY, WITH THE EXPLICIT COMCIDP     435    A
COMCIDP  *              RESULT REGISTER SPECIFIED.  E.G. --                     COMCIDP     436    A
COMCIDP  *                                                                      COMCIDP     437    A
COMCIDP  *                BX4  -X0*X5    (X4) = NEXT CHAR TO PACK, -R- FORMAT   COMCIDP     438    A
COMCIDP  *                                                                      COMCIDP     439    A
COMCIDP  *              WHEN SETTING A LOGICAL FLAG, IT DIFFERS --              COMCIDP     440    A
COMCIDP  *                                                                      COMCIDP     441    A
COMCIDP  *                SA6  TF=SQZ    SET TO *INDICATE NO BLANK SQUEEZE*      COMCIDP     442    A
COMCIDP  *                                                                      COMCIDP     443    A
COMCIDP  *              OR WHEN MANIPULATING A *COMADEF* STRUCTURE --           COMCIDP     444    A
COMCIDP  *                                                                      COMCIDP     445    A
COMCIDP  *                BX6  X6+X1     MERGE *LEN*                            COMCIDP     446    A
COMCIDP  *                                                                      COMCIDP     447    A
COMCIDP  *              IN ITS MOST COMMON FORM, THE RESULT REGISTER IS         COMCIDP     448    A
COMCIDP  *              EXPLICITLY SPECIFIED SO THAT THE READER KNOWS           COMCIDP     449    A
COMCIDP  *              **ABSOLUTELY** WHAT IS MEANT.                           COMCIDP     450    A
COMCIDP  *                                                                      COMCIDP     451    A
COMCIDP  *              IN ANY CASE, RESERVING COLS 30-72 FOR DATA STRUCTURE    COMCIDP     452    A
COMCIDP  *              DESCRIPTIONS MAKES DEBUGGING EASIER BECAUSE ONE CAN     COMCIDP     453    A
COMCIDP  *              CONSTANTLY COMPARE WHAT **IS** WITH WHAT **SHOULD BE**. COMCIDP     454    A
COMCIDP  *                                                                      COMCIDP     455    A
COMCIDP  *           2. FUNCTIONAL COMMENTS (I.E. WHAT IS GOING ON, OR HOW IT   COMCIDP     456    A
COMCIDP  *              IS DONE, OR MORE IMPORTANTLY, WHY ARE WE DOING IT) ARE  COMCIDP     457    A
COMCIDP  *              USUALLY PLACED ON A SEPARATE COMMENT LINE THAT PRECEDES COMCIDP     458    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *              PARAGRAPH OF CODE THAT IT PERTAINS TO.                  COMCIDP     459     A
COMCIDP  *                                                                      COMCIDP     460     A
COMCIDP  *              THIS TENDS TO MAKE CODE OCCUR IN FUNCTIONAL BLOCKS...GOOD COMCIDP   461     A
COMCIDP  *              FOR A QUICK ONCE-OVER EYE SCAN.                         COMCIDP     462     A
COMCIDP  *                                                                      COMCIDP     463     A
COMCIDP  *           3. A FLAG CELL DEFINITION WILL CONTAIN (AT A MINIMUM)      COMCIDP     464     A
COMCIDP  *              A COMMENT THAT SPECIFIED/DESCRIBES **ALL** POSSIBLE     COMCIDP     465     A
COMCIDP  *              VALUES THAT THIS FLAG MAY TAKE, AND IDEALLY, UNDER      COMCIDP     466     A
COMCIDP  *              WHAT CONDITIONS THE FLAG WILL TAKE THOSE VALUES.        COMCIDP     467     A
COMCIDP  *                                                                      COMCIDP     468     A
COMCIDP  *              THIS IS DONE BECAUSE OFTENTIMES, THE FLAG CELL DEFINITION COMCIDP   469     A
COMCIDP  *              IS THE ONLYPLACE WHERE THE INTRICACIES AND PATHOLOGIES  COMCIDP     470     A
COMCIDP  *              ASSOCIATED WITH THE USE OF THE FLAG CAN BE INTEGRATED.  COMCIDP     471     A
COMCIDP  *                                                                      COMCIDP     472     A
COMCIDP  ************************************************************************ COMCIDP  473     A
COMCIDP  *                                                                      COMCIDP     474     A
COMCIDP  *         SYMBOL NAMING                                                COMCIDP     475     A
COMCIDP  *                                                                      COMCIDP     476     A
COMCIDP  *         THIS IS A DESCRIPTION OF THE CONVENTIONS USED IN SYMBOL      COMCIDP     477     A
COMCIDP  *         NAMING.  THE INTENT OF THSE CONVENTIONS IS TO TRANSMIT TO    COMCIDP     478     A
COMCIDP  *         THE READER AS MUCH INFORMATION AS POSSIBLE BY MERELY SEEING  COMCIDP     479     A
COMCIDP  *         A SYMBOL NAME ITSELF.  OFTEN, THIS INFORMATION IMPLIES HOW   COMCIDP     480     A
COMCIDP  *         ONE CAN **EXPECT** A SYMBOL TO BE USED.                      COMCIDP     481     A
COMCIDP  *                                                                      COMCIDP     482     A
COMCIDP  *         TWO BASIC SYMBOL NAME-FORMING RULES ARE USED: FIRST, SYMBOL  COMCIDP     483     A
COMCIDP  *         PREFIXES ARE USED HEAVILY.  AND SECOND, THE CHARACTER THAT   COMCIDP     484     A
COMCIDP  *         SEPARATES A SYMBOL PREFIX FROM THE REST OF THE SYMBOL NAME   COMCIDP     485     A
COMCIDP  *         IS GIVEN SEMANTIC MEAING.  CURRENTLY, THERE ARE ONLY 2 SUCH  COMCIDP     486     A
COMCIDP  *         CHARACTERS: -.- IMPLIES THAT THE VALUE OF THE SYMBOL IS      COMCIDP     487     A
COMCIDP  *         AVAILABLE AT ASSEMBLY-TIME.  E.G. --                         COMCIDP     488     A
COMCIDP  *                                                                      COMCIDP     489     A
COMCIDP  *                SX1    L.KEYW  (X1) = LEN (IN WORDS) OF *KEYW* TABLE  COMCIDP     490     A
COMCIDP  *                                                                      COMCIDP     491     A
COMCIDP  *         AN -=- CHARACTER IMPLIES THAT THE VALUE OF THE SYMBOL IS     COMCIDP     492     A
COMCIDP  *         AVAILABLE ONLY AT EXECUTION-TIME (I.E. IT IS ONLY THE ADDRESS COMCIDP    493     A
COMCIDP  *         OF A CELL AT ASSEMBLY-TIME).  E.G. --                        COMCIDP     494     A
COMCIDP  *                                                                      COMCIDP     495     A
COMCIDP  *                SA1    L=KEYW  (X1) = LEN (IN WORDS) OF *KEYW* TABLE  COMCIDP     496     A
COMCIDP  *                                                                      COMCIDP     497     A
COMCIDP  *         THESE SYMBOLS ARE, THEREFORE, INDIRECT.                      COMCIDP     498     A
COMCIDP  *                                                                      COMCIDP     499     A
COMCIDP  *         NOTE THE USE OF THE WORD *VALUE* IN THE ABOVE PARAGRAPH.     COMCIDP     500     A
COMCIDP  *         OBVIOUSLY, *COMPASS* MUST BE ABLE TO ASSOCIATE SOME BINARY   COMCIDP     501     A
COMCIDP  *         NUMBER WITH BOTH TYPES OF SYMBOLS AT ASSEMBLY-TIME.  IN A    COMCIDP     502     A
COMCIDP  *         HUMAN SENSE, THOUGH, THE SYMBOL NAMES IN THE ABOVE EXAMPLES  COMCIDP     503     A
COMCIDP  *         TELL WHETHER THE *VALUE* OF THE SYMBOL WILL BE USED IN       COMCIDP     504     A
COMCIDP  *         A DIRECT, OR AN INDIRECT MANNER.  A SUBTLE DIFFERENCE THAT   COMCIDP     505     A
COMCIDP  *         CAN BE HANDY WHEN READING CODE.                              COMCIDP     506     A
COMCIDP  *                                                                      COMCIDP     507     A
COMCIDP  *         SYMBOL NAMES CAN BE DIVIDED INTO THE FOLLOWING GENERAL       COMCIDP     508     A
COMCIDP  *         TYPES --                                                     COMCIDP     509     A
COMCIDP  *                                                                      COMCIDP     510     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *          1. SYMBOLS THAT DESCRIBE COMMON COMPUTER OR CDC CONCEPTS,          COMCIDP    511    A
COMCIDP  *             SUCH AS: LENGTH, FIRST WORD ADDRESS, ETC...  THESE              COMCIDP    512    A
COMCIDP  *             SYMBOL NAMES ARE STRUCTURED SO THAT THE PREFIX DESCRIBES        COMCIDP    513    A
COMCIDP  *             THE COMMON IDEA.  I.E. -                                        COMCIDP    514    A
COMCIDP  *                                                                             COMCIDP    515    A
COMCIDP  *               L.XXX  - LENGTH OF XXX.                                       COMCIDP    516    A
COMCIDP  *               FW.XXX - FWA OF XXX.                                          COMCIDP    517    A
COMCIDP  *               F.XXX  - FWA OF A FET/FIT FOR FILE XXX.                       COMCIDP    518    A
COMCIDP  *                                                                             COMCIDP    519    A
COMCIDP  *          2. *COMADEF* STRUCTURES.  *COMADEF* IS A COMDECK THAT              COMCIDP    520    A
COMCIDP  *             CONTAINS MACROS FOR DESCRIBE/DEFINING SOFTWARE DATA             COMCIDP    521    A
COMCIDP  *             STRUCTURES.  CONCEPTUALLY, IT WORKS MUCH LIKE THE *COBOL*       COMCIDP    522    A
COMCIDP  *             *PICTURE*.  VIA THE *COMADEF* MACROS, ONE CAN DRAW              COMCIDP    523    A
COMCIDP  *             A PICTURE OF WHAT A DATA STRUCTURE ELEMENT WILL LOOK            COMCIDP    524    A
COMCIDP  *             LIKE.                                                           COMCIDP    525    A
COMCIDP  *                                                                             COMCIDP    526    A
COMCIDP  *             NOW WHEN ONE WISHES TO BUILD OR ACCESS SOME PART OF THE         COMCIDP    527    A
COMCIDP  *             DATA STRUCTURE, YOU CAN DO IT **SYMBOLICALLY** VIA              COMCIDP    528    A
COMCIDP  *             SYMBOLS THAT THE *COMADEF* MACROS DEFINED IN RESPONSE           COMCIDP    529    A
COMCIDP  *             TO THE PICTURE THAT WAS DRAWN VIA THOSE MACROS.                 COMCIDP    530    A
COMCIDP  *                                                                             COMCIDP    531    A
COMCIDP  *             *COMADEF* SYMBOLS TAKE THE FORM --                              COMCIDP    532    A
COMCIDP  *                                                                             COMCIDP    533    A
COMCIDP  *                 DD.XXXP  AND DD.XXXL                                        COMCIDP    534    A
COMCIDP  *                                                                             COMCIDP    535    A
COMCIDP  *             WHERE *DD* IS A COMMON PREFIX FOR THIS DATA STRUCTURE,          COMCIDP    536    A
COMCIDP  *             *XXX* IS A UNIQUE NAME FOR A FIELD WITHIN THE DATA              COMCIDP    537    A
COMCIDP  *             STRUCTURE *DD*, *P* IS A SUFFIX THAT DENOTES THE RIGHT-         COMCIDP    538    A
COMCIDP  *             MOST BIT POSITION OF FIELD *XXX* WITHIN *DD*, AND *L*           COMCIDP    539    A
COMCIDP  *             IS A SUFFIX THAT DENOTES THE LENGTH (IN BITS) OF FIELD          COMCIDP    540    A
COMCIDP  *             *XXX*.                                                          COMCIDP    541    A
COMCIDP  *                                                                             COMCIDP    542    A
COMCIDP  *             FOR EXAMPLE --                                                  COMCIDP    543    A
COMCIDP  *                                                                             COMCIDP    544    A
COMCIDP  *                 SA1  A1+B1    (X1) = NEXT ELEMENT IN STRUCTURE *BB.*        COMCIDP    545    A
COMCIDP  *                 MX2  -BB.TOTL (X2) = MASK FOR EXTRACTING *TOT* FIELD        COMCIDP    546    A
COMCIDP  *                 LX1  0-BB.TOTP RIGHT-JUSTIFY *TOT* FIELD                    COMCIDP    547    A
COMCIDP  *                 BX6  -X2*X1   (X6) = CONTENTS OF *TOT* FIELD                COMCIDP    548    A
COMCIDP  *                                                                             COMCIDP    549    A
COMCIDP  *             ONCE ONE BECOMES FAMILIAR WITH THE *COMADEF* NOTATION,          COMCIDP    550    A
COMCIDP  *             MOST OF THE ABOVE COMMENTS ARE UNNECESSARY.  THE                COMCIDP    551    A
COMCIDP  *             ACCESSING OF A *COMADEF* STRUCTURE IS SO STYLIZED THAT          COMCIDP    552    A
COMCIDP  *             ONE CAN PRETTY MUCH RECOGNIZE IT ANYWHERE, BY ANYONE,           COMCIDP    553    A
COMCIDP  *             AND IN AN INFINITY OF CIRCUMSTANCES.                            COMCIDP    554    A
COMCIDP  *                                                                             COMCIDP    555    A
COMCIDP  *          3. SYMBOLS THAT ONE WISHES TO ASSOCIATE VIA A COMMON PREFIX.       COMCIDP    556    A
COMCIDP  *             THESE ARE SYMBOLS THAT THE PROGRAMMER FEELS HAVE AN             COMCIDP    557    A
COMCIDP  *             INTERRELATIONSHIP WITHIN THE CONTEXT OF A SPECIFIC              COMCIDP    558    A
COMCIDP  *             PIECE OF CODE.  FOR EXAMPLE, ALL SYMBOL TYPES (DECK NAME,       COMCIDP    559    A
COMCIDP  *             *SET* NAME, ETC) ARE DEFINED VIA SYMBOLS THAT HAVE THE          COMCIDP    560    A
COMCIDP  *             THE COMMON PREFIX *SY.*, E.G. *SY.DECK*, *SY.SET*, ETC.         COMCIDP    561    A
COMCIDP  *                                                                             COMCIDP    562    A

             0         1         2         3         4         5         6         7         8
             12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *              CERTAIN GROUPS OF DATA CELLS ARE ALSO ASSOCIATED VIA          COMCIDP    563    A
COMCIDP  *              A COMMON PREFIX: E.G. ALL THE CELLS THAT CONTAIN              COMCIDP    564    A
COMCIDP  *              INFORMATION THAT *IDP* HAS GATHERED ABOUT THE TOKEN           COMCIDP    565    A
COMCIDP  *              BUFFER HAVE THE COMMON PREFIX *TB=*.                          COMCIDP    566    A
COMCIDP  *                                                                            COMCIDP    567    A
COMCIDP  *           4. SYMBOLS WITHIN MACRO DEFINITIONS.  THE CONVENTION             COMCIDP    568    A
COMCIDP  *              USED FOR NAMING SYMBOLS WITHIN A MACRO DEFINITION IS          COMCIDP    569    A
COMCIDP  *              AS FOLLOWS.  SYMBOLS ARE OF THE FORM --                       COMCIDP    570    A
COMCIDP  *                                                                            COMCIDP    571    A
COMCIDP  *                '?PPTXXX                                                    COMCIDP    572    A
COMCIDP  *                                                                            COMCIDP    573    A
COMCIDP  *              WHERE '?  = COMMON *MACRO SYMBOL* PREFIX.                     COMCIDP    574    A
COMCIDP  *                    PP  = A COMMON PREFIX FOR SYMBOLS WITHIN THIS           COMCIDP    575    A
COMCIDP  *                          MACRO DEFINITION, OR WITHIN A GROUP OF            COMCIDP    576    A
COMCIDP  *                          RELATED MACRO DEFINITIONS.                        COMCIDP    577    A
COMCIDP  *                    T   = SYMBOL TYPE.  SEE BELOW.                          COMCIDP    578    A
COMCIDP  *                    XXX = UNIQUE NAME FOR THIS SYMBOL.                      COMCIDP    579    A
COMCIDP  *                                                                            COMCIDP    580    A
COMCIDP  *              SYMBOL TYPES, *T*, ARE USED TO SIMPLIFY THE READING           COMCIDP    581    A
COMCIDP  *              OF MACRO CODE A LITTLE BIT.  EACH SYMBOL TYPE DENOTES         COMCIDP    582    A
COMCIDP  *              HOW THE SYMBOL WILL BE USED WITHIN THE MACRO.                 COMCIDP    583    A
COMCIDP  *                                                                            COMCIDP    584    A
COMCIDP  *              SYMBOL TYPES ARE --                                           COMCIDP    585    A
COMCIDP  *                                                                            COMCIDP    586    A
COMCIDP  *                S = *SET* SYMBOL, I.E. ANY SYMBOL THAT IS ASSIGNED AN       COMCIDP    587    A
COMCIDP  *                    ABSOLUTE VALUE.  E.G. '?PPSXXX SET 1                    COMCIDP    588    A
COMCIDP  *                                                                            COMCIDP    589    A
COMCIDP  *                C = *MICRO* NAME.  E.G. '?PPCXXX MICRO 1,,/HARPIES/         COMCIDP    590    A
COMCIDP  *                                                                            COMCIDP    591    A
COMCIDP  *                R = *RMT* BLOCK NAME.  E.G. '?PPRXXX RMT                    COMCIDP    592    A
COMCIDP  *                                                                            COMCIDP    593    A
COMCIDP  *                M = *MACRO* NAME.  I.E. USED FOR INTERNAL MACROS WHICH      COMCIDP    594    A
COMCIDP  *                    THE PROGRAMMER WISHES TO BE **INVISIBLE** TO THE        COMCIDP    595    A
COMCIDP  *                    REST OF THE WORLD.                                      COMCIDP    596    A
COMCIDP  *                                                                            COMCIDP    597    A
COMCIDP  *           5. NONE OF THE ABOVE.  THERE WILL ALWAYS BE EXCEPTIONS...        COMCIDP    598    A
COMCIDP  *              SOME OF THESE ARE BECAUSE THERE IS NO CONVENTION, SOME        COMCIDP    599    A
COMCIDP  *              ARE BECAUSE THE CONVENTION DIDNT SEEM RIGHT FOR THIS          COMCIDP    600    A
COMCIDP  *              PARTICULAR CASE, ETC, ETC, ETC...                             COMCIDP    601    A
COMCIDP  *                                                                            COMCIDP    602    A
COMCIDP  ***********************************************************************     COMCIDP    603    A
COMCIDP  *                                                                            COMCIDP    604    A
COMCIDP  *         GLOSSARY                                                           COMCIDP    605    A
COMCIDP  *                                                                            COMCIDP    606    A
COMCIDP  *         THE FOLLOWING IS A GLOSSARY OF COMMON ABBREVIATIONS,               COMCIDP    607    A
COMCIDP  *         NOTATIONS, AND SYMBOL PREFIXES THAT ARE NOT IN PARTICULAR          COMCIDP    608    A
COMCIDP  *         *IDP* DEPENDENT.                                                   COMCIDP    609    A
COMCIDP  *                                                                            COMCIDP    610    A
COMCIDP  *         A+C    ADDRESS AND CONTENTS.  USED PRIMARILY IN CONJUNCTION        COMCIDP    611    A
COMCIDP  *                WITH A LOAD INSTRUCTION WHERE BOTH THE ADDRESS AND          COMCIDP    612    A
COMCIDP  *                THE CONTENTS OF THE ADDRESS ARE MEANINGFUL, AS IN --        COMCIDP    613    A
COMCIDP  *                                                                            COMCIDP    614    A

         0        1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *                  SA1   A1+B1   (A1,X1) = A+C OF NEXT TABLE ENTRY          COMCIDP     615     A
COMCIDP  *                                                                            COMCIDP     616     A
COMCIDP  *               AS OPPOSED TO --                                             COMCIDP     617     A
COMCIDP  *                                                                            COMCIDP     618     A
COMCIDP  *                  SA1   LN=TYPE (X1) = LINE TYPE                            COMCIDP     619     A
COMCIDP  *                                                                            COMCIDP     620     A
COMCIDP  *         ADDR   ADDRESS.                                                    COMCIDP     621     A
COMCIDP  *                                                                            COMCIDP     622     A
COMCIDP  *         FWA    FIRST WORD ADDRESS.                                         COMCIDP     623     A
COMCIDP  *                                                                            COMCIDP     624     A
COMCIDP  *         F.XXX  FWA OF A LOGICAL FILE *FET*.                                COMCIDP     625     A
COMCIDP  *                                                                            COMCIDP     626     A
COMCIDP  *         FW=XXX A CELL THAT CONTAINS A FWA.                                 COMCIDP     627     A
COMCIDP  *                                                                            COMCIDP     628     A
COMCIDP  *         FW.XXX A SYMBOL WHOSE VALUE SPECIFIES A FWA.                       COMCIDP     629     A
COMCIDP  *                                                                            COMCIDP     630     A
COMCIDP  *         L=XXX  A CELL THAT CONTAINS A LENGTH (NOT A MANAGED TABLE          COMCIDP     631     A
COMCIDP  *                *LENGTH*, HOWEVER).                                         COMCIDP     632     A
COMCIDP  *                                                                            COMCIDP     633     A
COMCIDP  *         L.XXX  A SYMBOL WHOSE VALUE SPECIFIES A LENGTH IN WORDS.           COMCIDP     634     A
COMCIDP  *                                                                            COMCIDP     635     A
COMCIDP  *         L1.XXX A SYMBOL WHOSE VALUE SPECIFIES THE LENGTH (IN WORDS)        COMCIDP     636     A
COMCIDP  *                OF A SINGLE TABLE ENTRY.  FOR EXAMPLE --                    COMCIDP     637     A
COMCIDP  *                                                                            COMCIDP     638     A
COMCIDP  *                  SA1   FW.TBL    (A1,X1) = A+C OF 1ST *TBL* ENTRY          COMCIDP     639     A
COMCIDP  *                  SA2   A1+L1.TBL (A2,X2) = A+C OF 2ND *TBL* ENTRY          COMCIDP     640     A
COMCIDP  *                                                                            COMCIDP     641     A
COMCIDP  ************************************************************************   COMCIDP     642     A
COMCIDP   IDP      TITLE  IDP "IDPVER"/DATA STRUCTURES.                               COMCIDP     643     A
COMCIDP   ADR=RJ   SPACE  4,10                                                        COMCIDP     644     A
COMCIDP  **        ADR=RJ - ADDRESS OF CALLING RJ.                                    COMCIDP     645     A
COMCIDP  *                                                                            COMCIDP     646     A
COMCIDP  *                                                                            COMCIDP     647     A
COMCIDP  *         *ADR=RJ* CONTAINS THE ADDRESS OF THE *RJ* INSTRUCTION THAT         COMCIDP     648     A
COMCIDP  *         LAST INVOKED ONE OF THE EXECUTIVES (IDP=,REG=,SNP=) IN             COMCIDP     649     A
COMCIDP  *         *COMCIDP*.  IT IS SET UP BY *IEX* (INITIALIZE EXECUTIVE),          COMCIDP     650     A
COMCIDP  *         AND IS USED WHENEVER SOMEONE WISHES TO KNOW THE ADDR OF            COMCIDP     651     A
COMCIDP  *         AN IDP CALLER.  E.G. *HDR* (OUTPUT HEADER) USES (ADR=RJ)           COMCIDP     652     A
COMCIDP  *         IN OUTPUTING THE *CALLED BY*-TYPE HEADERS.                         COMCIDP     653     A
COMCIDP                                                                               COMCIDP     654     A
COMCIDP                                                                               COMCIDP     655     A
COMCIDP   ADR=RJ   BSSZ   1                                                           COMCIDP     656     A
COMCIDP   APL      SPACE  4,10                                                        COMCIDP     657     A
COMCIDP  **        APL - IDP APLIST.                                                  COMCIDP     658     A
COMCIDP  *                                                                            COMCIDP     659     A
COMCIDP  *                                                                            COMCIDP     660     A
COMCIDP  *         *APL* IS AN AREA USED FOR 2 PURPOSES --                            COMCIDP     661     A
COMCIDP  *                                                                            COMCIDP     662     A
COMCIDP  *           1. WHEN IDP IS INVOKED VIA AN INTERACTIVE BREAK (I.E. NOT        COMCIDP     663     A
COMCIDP  *              AN ASSEMBLED BREAK), THEN *BRK* (BREAKPOINT PROCESSOR)        COMCIDP     664     A
COMCIDP  *              EXPANDS ITS INTERNAL BREAKPOINT INFORMATION FROM *IDPBA*      COMCIDP     665     A
COMCIDP  *              (BREAKPOINT ADDRESS TABLE) AND *IDPBC* (BREAKPOINT            COMCIDP     666     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *              CONTENTS TABLE) INTO *APL* ET SEQ SO THAT THIS                COMCIDP     667    A
COMCIDP  *              INFORMATION **LOOKS** JUST LIKE AN ASSEMBLED BREAKPOINT       COMCIDP     668    A
COMCIDP  *              PARAMETER LIST.                                               COMCIDP     669    A
COMCIDP  *                                                                            COMCIDP     670    A
COMCIDP  *           2. AND BECAUSE THIS *BREAKPOINT INFORMATION* IS NOT NEEDED       COMCIDP     671    A
COMCIDP  *              FOR VERY LONG, IDP REUSES THIS AREA FOR SCRATCH CELLS.        COMCIDP     672    A
COMCIDP                                                                               COMCIDP     673    A
COMCIDP                                                                               COMCIDP     674    A
COMCIDP   L.APL    =      SN=LEN+1                                                    COMCIDP     675    A
COMCIDP   APL      BSSZ   L.APL                                                       COMCIDP     676    A
COMCIDP                                                                               COMCIDP     677    A
COMCIDP            LOC    APL                                                         COMCIDP     678    A
COMCIDP                                                                               COMCIDP     679    A
COMCIDP   AP=LL    =      *+SN=LL                                                     COMCIDP     680    A
COMCIDP   AP=UL    =      *+SN=UL                                                     COMCIDP     681    A
COMCIDP   AP=INC   =      *+SN=INC                                                    COMCIDP     682    A
COMCIDP   AP=HDR   =      *+SN=HDR                                                    COMCIDP     683    A
COMCIDP   AP=CNT   =      *+SN=CNT                                                    COMCIDP     684    A
COMCIDP   AP=FWA   =      *+SN=FWA                                                    COMCIDP     685    A
COMCIDP   AP=LWA   =      *+SN=LWA                                                    COMCIDP     686    A
COMCIDP   AP=LEN   =      *+SN=LEN                                                    COMCIDP     687    A
COMCIDP            LOC    *O                                                          COMCIDP     688    A
COMCIDP   BC=BRAD  SPACE  4,10                                                        COMCIDP     689    A
COMCIDP   .IDPOS   IFNE   .OS,2                                                       CCGA018       5    A
COMCIDP  **        BC=BRAD - PREVIOUS CONTENTS OF BREAK ADDRESS.                      COMCIDP     690    A
COMCIDP  *                                                                            COMCIDP     691    A
COMCIDP  *                                                                            COMCIDP     692    A
COMCIDP                                                                               COMCIDP     693    A
COMCIDP                                                                               COMCIDP     694    A
COMCIDP   BC=BRAD  BSSZ   1                                                           COMCIDP     695    A
COMCIDP   F.       SPACE  4,10                                                        COMCIDP     696    A
COMCIDP  ***       F. - FILE FETS.                                                    COMCIDP     697    A
COMCIDP  *                                                                            COMCIDP     698    A
COMCIDP  *                                                                            COMCIDP     699    A
COMCIDP  *         SYMBOLS OF THE FORM *F.XXX*, WHERE *F.* IS A COMMON PREFIX,        COMCIDP     700    A
COMCIDP  *         AND *XXX* IS A UNIQUE FILE, DEFINE THE FWA OF A FET USED           COMCIDP     701    A
COMCIDP  *         BY *IDP*.                                                          COMCIDP     702    A
COMCIDP  *                                                                            COMCIDP     703    A
COMCIDP  *         FOLLOWING ARE ALL THE DEFINITIONS OF FETS USED BY *IDP*,           COMCIDP     704    A
COMCIDP  *         WITH THE **EXCEPTION** OF *F.FRZ*, THE *FREEZE* FILE               COMCIDP     705    A
COMCIDP  *         (SEE *IFR*, IDP FREEZE RESTART).                                   COMCIDP     706    A
COMCIDP  *                                                                            COMCIDP     707    A
COMCIDP  ***************************************************************             COMCIDP     708    A
COMCIDP   F.BDO    SPACE  4,10                                                        COMCIDP     709    A
COMCIDP  **        F.BDO - FET AND BUFFER FOR BATCH DEBUG OUTPUT FILE.                COMCIDP     710    A
COMCIDP                                                                               COMCIDP     711    A
COMCIDP   #BDO     IF     -DEF,F.BDO                                                  COMCIDP     712    A
COMCIDP   L.BDO    =      101B        LENGTH OF BATCH OUTPUT BUFFER                   COMCIDP     713    A
COMCIDP                                                                               COMCIDP     714    A
COMCIDP   F.BDO    BSS    0           ** FWA OF FET **                                COMCIDP     715    A
COMCIDP   BDO      FILEC  IDPBDO,L.BDO                                                COMCIDP     716    A
COMCIDP   IDPBDO   BSS    L.BDO                                                       COMCIDP     717    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP   #BDO     ENDIF                                                    COMCIDP      718     A
COMCIDP   F.IDI    SPACE  4,10                                              COMCIDP      719     A
COMCIDP  **        F.IDI - FET, LINE IMAGE AREA, AND BUFFER FOR INTERACTIVE COMCIDP      720     A
COMCIDP  *           DEBUG INPUT FILE.                                      COMCIDP      721     A
COMCIDP                                                                     COMCIDP      722     A
COMCIDP   #IDI     IF     -DEF,F.IDI                                        COMCIDP      723     A
COMCIDP   L.IDI    =      101B        LENGTH OF *IDP* INPUT BUFFER          COMCIDP      724     A
COMCIDP                                                                     COMCIDP      725     A
COMCIDP   F.IDI    BSS    0           ** FWA OF FET **                      COMCIDP      726     A
COMCIDP   ZZZZZDI  FILEC  IDPIDI,L.IDI,(FET=7)                              COMCIDP      727     A
COMCIDP            DIS    1, **IDP**                                        COMCIDP      728     A
COMCIDP   FW.LIM   DIS    8,INTERACTIVE DEBUG LINE IMAGE AREA.              COMCIDP      729     A
COMCIDP   IDPIDI   BSS    L.IDI                                             COMCIDP      730     A
COMCIDP                                                                     COMCIDP      731     A
COMCIDP   L=LIM    BSSZ   1           NR OF WORDS IN SOURCE LINE IMAGE AT  COMCIDP      732     A
COMCIDP  *                              (FW.LIM) ET SEQ (INCLUDES WORD CONTAINING COMCIDP 733     A
COMCIDP  *                              EOL MARK)                           COMCIDP      734     A
COMCIDP   #IDI     ENDIF                                                    COMCIDP      735     A
COMCIDP   F.BDI    SPACE  4,10                                              COMCIDP      736     A
COMCIDP   F.BDI    =      F.IDI                                             COMCIDP      737     A
COMCIDP   F.IDO    SPACE  4,10                                              COMCIDP      738     A
COMCIDP  **        F.IDO - FET AND BUFFER FOR INTERACTIVE DEBUG OUTPUT FILE. COMCIDP     739     A
COMCIDP                                                                     COMCIDP      740     A
COMCIDP   #IDO     IF     -DEF,F.IDO                                        COMCIDP      741     A
COMCIDP   L.IDO    =      101B        LENGTH OF *IDP* OUTPUT BUFFER         COMCIDP      742     A
COMCIDP                                                                     COMCIDP      743     A
COMCIDP   F.IDO    BSS    0           ** FWA OF FET **                      COMCIDP      744     A
COMCIDP   ZZZZZDO  FILEC  IDPIDO,L.IDO,(FET=7)                              COMCIDP      745     A
COMCIDP   IDPIDO   BSS    L.IDO                                             COMCIDP      746     A
COMCIDP   #IDO     ENDIF                                                    COMCIDP      747     A
COMCIDP   .IDPOS   ENDIF                                                    CCGA018        6     A
COMCIDP   FW.SVR   SPACE  4,10                                              COMCIDP      748     A
COMCIDP  **        FW.SVR - SAVED REGISTER AREA.                            COMCIDP      749     A
COMCIDP  *                                                                  COMCIDP      750     A
COMCIDP  *                                                                  COMCIDP      751     A
COMCIDP  *         THIS IS WHERE IDP SAVES THE USERS REGISTERS.             COMCIDP      752     A
COMCIDP                                                                     COMCIDP      753     A
COMCIDP                                                                     COMCIDP      754     A
COMCIDP   FW.SVR   BSS    0           ** FWA **                             COMCIDP      755     A
COMCIDP   IDP=SVB  BSSZ   8           SAVED -B- REGISTERS                   COMCIDP      756     A
COMCIDP   IDP=SVA  BSSZ   8           SAVED -A- REGISTERS                   COMCIDP      757     A
COMCIDP   IDP=SVX  BSSZ   8           SAVED -X- REGISTERS                   COMCIDP      758     A
COMCIDP   FW=PARM  SPACE  4,10                                              COMCIDP      759     A
COMCIDP  **        FW=PARM - FWA OF BREAK/SNAP/REG PARAMETER LIST.          COMCIDP      760     A
COMCIDP  *                                                                  COMCIDP      761     A
COMCIDP  *                                                                  COMCIDP      762     A
COMCIDP                                                                     COMCIDP      763     A
COMCIDP                                                                     COMCIDP      764     A
COMCIDP   FW=PARM  BSSZ   1           SAVE CELL FOR FWA OF PARAMETER LIST   COMCIDP      765     A
COMCIDP   IDPERR   SPACE  4,10                                              COMCIDP      766     A
COMCIDP  **        IDPERR - IDP ERROR CELL.                                 COMCIDP      767     A
COMCIDP  *                                                                  COMCIDP      768     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *                                                                  COMCIDP     769    A
COMCIDP  *         *IDPERR* IS SET BY *IDP=ER* WHEN AN ERROR IS DETECTED.   COMCIDP     770    A
COMCIDP  *         ITS VALUE DEPENDS ON THE IDP ASSEMBLY-TIME OPTION *OI.ERR* --  COMCIDP  771  A
COMCIDP  *                                                                  COMCIDP     772    A
COMCIDP  *           1. IF *OI.ERR = 0*, THEN (IDPERR) ARE ALWAYS SET TO .ZR.  COMCIDP  773    A
COMCIDP  *              TO INDICATE THAT NO ERROR MSG IS AVAILABLE.         COMCIDP     774    A
COMCIDP  *                                                                  COMCIDP     775    A
COMCIDP  *           2. IF *OI.ERR = 1*, THEN (IDPERR) ARE SET TO THE ERROR MSG  COMCIDP  776  A
COMCIDP  *              NR FOR THIS ERROR.                                  COMCIDP     777    A
COMCIDP  *                                                                  COMCIDP     778    A
COMCIDP  *           3. IF *OI.ERR = 2*, THEN (IDPERR) ARE SET TO THE FWA OF THE  COMCIDP  779  A
COMCIDP  *              ERROR MSG ITSELF.                                   COMCIDP     780    A
COMCIDP  *                                                                  COMCIDP     781    A
COMCIDP  *         SEE *OI.ERR* (IN *COMSIDP*) FOR MORE INFORMATION.        COMCIDP     782    A
COMCIDP                                                                     COMCIDP     783    A
COMCIDP                                                                     COMCIDP     784    A
COMCIDP   IDPERR   BSSZ   1                                                 COMCIDP     785    A
COMCIDP   IDPFLG   SPACE  4,10                                              COMCIDP     786    A
COMCIDP  **        IDPFLG - INTERACTIVE/BATCH MASTER CONTROL FLAG.          COMCIDP     787    A
COMCIDP  *                                                                  COMCIDP     788    A
COMCIDP  *               CONTAINS GLOBAL CONTROL INFORMATION THAT IS USED   COMCIDP     789    A
COMCIDP  *         THROUGH OUT *COMCIDP*.  THE FIELDS IN *IDPFLG* ARE DESCRIBED  COMCIDP  790  A
COMCIDP  *         BY PAIRS OF SYMBOLS OF THE FORM *IDF.XXXP* AND *IDF.XXXL*,  COMCIDP  791    A
COMCIDP  *         WHERE *IDF* IS THE COMMON PREFIX, *XXX* IS THE FIELD     COMCIDP     792    A
COMCIDP  *         DESCRIPTOR, *P* DENOTES THE RIGHT-MOST BIT OF THE FIELD, AND  COMCIDP  793  A
COMCIDP  *         *L* DENOTES THE LENGTH OF THE FIELD.                     COMCIDP     794    A
COMCIDP  *                                                                  COMCIDP     795    A
COMCIDP  *               FIELD DESCRIPTIONS (XXX) ARE --                    COMCIDP     796    A
COMCIDP  *                                                                  COMCIDP     797    A
COMCIDP  *         ADR  = 0  IF ADDRESSES ARE TO BE OUTPUT AS ABSOLUTE      COMCIDP     798    A
COMCIDP  *              = 1  IF ADDRESSES ARE TO BE OUTPUT AS DECK RELATIVE COMCIDP     799    A
COMCIDP  *                                                                  COMCIDP     800    A
COMCIDP  *         BDO  = 1  IF WRITING TO BATCH DEBUG OUTPUT FILE (F.BDO), COMCIDP     801    A
COMCIDP  *                     ELSE 0                                       COMCIDP     802    A
COMCIDP  *                                                                  COMCIDP     803    A
COMCIDP  *         BKO  = 1  IF ECHOING/WRITING INPUT SOURCE LINE IMAGE TO  COMCIDP     804    A
COMCIDP  *                     BATCH DEBUG OUTPUT FILE *F.BDO*, ELSE 0      COMCIDP     805    A
COMCIDP  *                                                                  COMCIDP     806    A
COMCIDP  *         BRF  = 1  IF IN *BRIEF* MODE (I.E. DO NOT ISSUE INTERACTIVE  COMCIDP  807  A
COMCIDP  *                     PROMPT), ELSE 0.                             COMCIDP     808    A
COMCIDP  *                                                                  COMCIDP     809    A
COMCIDP  *         FTO  = 0  IF THIS IS 1ST TIME *IDP* HAS BEEN CALLED, ELSE 1  COMCIDP  810  A
COMCIDP  *                                                                  COMCIDP     811    A
COMCIDP  *         IDO  = 1  IF WRITING TO INTERACTIVE DEBUG OUTPUT FILE (F.IDO),  COMCIDP  812  A
COMCIDP  *                     ELSE 0                                       COMCIDP     813    A
COMCIDP  *                                                                  COMCIDP     814    A
COMCIDP  *         IKO  = 1  IF ECHOING/WRITING INPUT SOURCE LINE IMAGE TO  COMCIDP     815    A
COMCIDP  *                     INTERACTIVE DEBUG OUTPUT FILE *F.IDO*, ELSE 0  COMCIDP  816    A
COMCIDP  *                                                                  COMCIDP     817    A
COMCIDP  *         INP  = 1  IF READING FROM BATCH DEBUG INPUT FILE (F.BDI) COMCIDP     818    A
COMCIDP  *                     (I.E. READING FROM AN UNCONNECTED INPUT FILE)  COMCIDP  819    A
COMCIDP  *              = 0  IF READING FROM INTERACTIVE DEBUG INPUT FILE (F.IDI)  COMCIDP  820  A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *                                                          COMCIDP    821    A
COMCIDP  *        MEM  = 0  IF ALL MEMORY REFERENCES ARE TO BE CHECKED  COMCIDP    822    A
COMCIDP  *                  IN THE *NORMAL* WAY, ELSE                 COMCIDP    823    A
COMCIDP  *                                                            COMCIDP    824    A
COMCIDP  *             = 1  IF ALL MEMORY REFERENCES ARE TO BE CHECKED  COMCIDP    825    A
COMCIDP  *                  VIA A SYSTEM *MEM/MEMORY* CALL.  THIS OPTION  COMCIDP    826    A
COMCIDP  *                  IS INTENDED FOR THOSE TIMES WHEN IDP IS MODING  COMCIDP    827    A
COMCIDP  *                  OUT BECAUSE THE HOST MEMORY MANAGEMENT IS  COMCIDP    828    A
COMCIDP  *                  SCREWING UP (I.E. TELLING IDP THAT OUT-OF-RANGE ADDRS  COMCIDP    829    A
COMCIDP  *                  ARE OK), PUTTING THE USER UP SHIT CREEK SO TO SPEAK.  COMCIDP    830    A
COMCIDP  *                                                            COMCIDP    831    A
COMCIDP  *                  WHEN *MEM* IS SET, IDP WILL IGNORE ALL HOST MEMORY  COMCIDP    832    A
COMCIDP  *                  MANAGEMENT SCHEMES, RELYING SOLEY UPON THE OPERATING  COMCIDP    833    A
COMCIDP  *                  SYSTEM FOR FL INFORMATION.                COMCIDP    834    A
COMCIDP  *                                                            COMCIDP    835    A
COMCIDP  *        RAP  = 1  IF A CIO READ WAS ALREADY PERFORMED       COMCIDP    836    A
COMCIDP  *             = 0  IF A CIO READ NEEDS TO BE PERFORMED       COMCIDP    837    A
COMCIDP  *                                                            COMCIDP    838    A
COMCIDP  *        SNL  = NR OF CM WORDS TO BE DUMPED (BY *DCM*) ON A SINGLE LINE  COMCIDP    839    A
COMCIDP  *             = L.BSL  IN BATCH MODE                         COMCIDP    840    A
COMCIDP  *             = L.ISL  IN INTERACTIVE MODE                   COMCIDP    841    A
COMCIDP  *                                                            COMCIDP    842    A
COMCIDP  *        XEC  = 0  IF THE EXECUTIVE IS *IDP=*                COMCIDP    843    A
COMCIDP  *             = 1  IF THE EXECUTIVE IS *REG=*                COMCIDP    844    A
COMCIDP  *             = 2  IF THE EXECUTIVE IS *SNP=*                COMCIDP    845    A
COMCIDP  *                                                            COMCIDP    846    A
COMCIDP  *        XJR  = 1  IF *COMCXJR* IS TO BE USED FOR RESTORING REGISTERS,  COMCIDP    847    A
COMCIDP  *                  ELSE                                      COMCIDP    848    A
COMCIDP  *             = 0  IF *COMCRSR* IS TO BE USED FOR RESTORING REGISTERS.  COMCIDP    849    A
COMCIDP  *                                                            COMCIDP    850    A
COMCIDP  *                  SEE *OPTION,XJR* COMMAND IN IDP EXTERNAL REFERENCE  COMCIDP    851    A
COMCIDP  *                  MANUAL FOR EXAMPLE OF USE.                COMCIDP    852    A
COMCIDP                                                               COMCIDP    853    A
COMCIDP                                                               COMCIDP    854    A
COMCIDP   IDPFLG   BSSZ   1                                           COMCIDP    855    A
COMCIDP   MX=      SPACE  4,10                                        COMCIDP    856    A
COMCIDP   .IDPOS   IFNE   .OS,2                                       CCGA018      7    A
COMCIDP  **        MX= - MASKS.                                       COMCIDP    857    A
COMCIDP  *                                                            COMCIDP    858    A
COMCIDP  *                                                            COMCIDP    859    A
COMCIDP                                                               COMCIDP    860    A
COMCIDP                                                               COMCIDP    861    A
COMCIDP   '?IDCS01 CHMIC  (B,A,X)                                     COMCIDP    862    A
COMCIDP   MX=BAX   LIT    "'?IDCS01"                                  COMCIDP    863    A
COMCIDP                                                               COMCIDP    864    A
COMCIDP   '?IDCS01 CHMIC  (0,1,2,3,4,5,6,7)                           COMCIDP    865    A
COMCIDP   MX=0..7  LIT    "'?IDCS01"                                  COMCIDP    866    A
COMCIDP   MX=PREUN SPACE  4,10                                        COMCIDP    867    A
COMCIDP  **        MX=PREUN - PRE-UNARY TOKEN TYPE SHIFT MASK.        COMCIDP    868    A
COMCIDP  *                                                            COMCIDP    869    A
COMCIDP  *                                                            COMCIDP    870    A
COMCIDP  *         *MX=PREUN* IS A SHIFT MASK USED BY *PAS* (PARSE SUBEXPRESSION)  COMCIDP    871    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *         FOR DETERMINING WHETHER AN OPERATOR IS UNARY OR NOT.                COMCIDP     872    A
COMCIDP  *                                                                             COMCIDP     873    A
COMCIDP  *         FOR EXAMPLE --                                                      COMCIDP     874    A
COMCIDP  *                                                                             COMCIDP     875    A
COMCIDP  *               SNAP -A     IS UNARY                                          COMCIDP     876    A
COMCIDP  *               SNAP A-B    IS BINARY                                         COMCIDP     877    A
COMCIDP                                                                                COMCIDP     878    A
COMCIDP                                                                                COMCIDP     879    A
COMCIDP   '?IDCS01 CHMIC  (O.VAR,O.CONS,O.REG,O.RP)                                    COMCIDP     880    A
COMCIDP   MX=PREUN LIT    -"'?IDCS01"                                                  COMCIDP     881    A
COMCIDP   MX=TET   SPACE  4,10                                                         COMCIDP     882    A
COMCIDP  **        MX=TET - TERMINATING TOKEN TYPE SHIFT MASK.                         COMCIDP     883    A
COMCIDP  *                                                                             COMCIDP     884    A
COMCIDP  *                                                                             COMCIDP     885    A
COMCIDP  *         *MX=TET* CONTAINS A SHIFT MASK THAT IS USED BY *PAS* (PARSE         COMCIDP     886    A
COMCIDP  *         SUBEXPRESSION) TO DETERMINE WHAT TOKEN TYPES (*TOT*) ARE TO         COMCIDP     887    A
COMCIDP  *         TERMINATE THE PARSE.                                                COMCIDP     888    A
COMCIDP  *                                                                             COMCIDP     889    A
COMCIDP  *         FOR MOST STMTS, THEY ARE THE *EOS* (END-OF-STMT) AND *COM*          COMCIDP     890    A
COMCIDP  *         (COMMA) TOKENS, BUT BECAUSE THIS MAY CHANGE SOME DAY (FUNCTION      COMCIDP     891    A
COMCIDP  *         REFERENCES, FOR EXAMPLE), *PAS* USES THIS SHIFT MASK.               COMCIDP     892    A
COMCIDP  *                                                                             COMCIDP     893    A
COMCIDP  *         NOTE THAT *PAS* INTERNALLY USES (PAS=TET) FOR ITS TERMINATING       COMCIDP     894    A
COMCIDP  *         *TOT* SHIFT MASK, BUT **GUARANTEES** THAT, UPON EXIT,               COMCIDP     895    A
COMCIDP  *         (PAS=TET) IS SET TO (MX=TET).  THIS RELATIONSHIP CAN BE             COMCIDP     896    A
COMCIDP  *         THOUGHT OF AS A MASTER/SLAVE, AND ALLOWS A CALLER TO ALTER          COMCIDP     897    A
COMCIDP  *         THE WAY *PAS* PARSES, BUT DOES NOT REQUIRE THAT ALL THE             COMCIDP     898    A
COMCIDP  *         RUN-OF-THE-MILL CALLERS SET UP (PAS=TET).                           COMCIDP     899    A
COMCIDP  *                                                                             COMCIDP     900    A
COMCIDP  *         NOTE HOW THE MICRO *'?IDCTET* IS USED TO INITIALLY SET UP           COMCIDP     901    A
COMCIDP  *         **BOTH** (MX=TET) AND (PAS=TET).                                    COMCIDP     902    A
COMCIDP  *                                                                             COMCIDP     903    A
COMCIDP  *         I CONSIDER IT A SAFE PRACTICE TO PLACE *BITMIC* PARAMETERS          COMCIDP     904    A
COMCIDP  *         IN **ASCENDING** ORDER...THIS AVOIDS A PROBLEM THAT CAN OCCUR       COMCIDP     905    A
COMCIDP  *         BECAUSE COMPASS SOMETIMES DOESN'T HONOR MIXED UP *POS*ING           COMCIDP     906    A
COMCIDP  *         (IT APPARENTLY HAS SOMETHING TO DO WITH WHEN COMPASS ACTUALLY       COMCIDP     907    A
COMCIDP  *         FORCES OUT BINARY WHEN ASSEMBLING PARTIAL WORDS).                   COMCIDP     908    A
COMCIDP                                                                                COMCIDP     909    A
COMCIDP                                                                                COMCIDP     910    A
COMCIDP   '?IDCTET CHMIC  (O.EOS,O.COM)                                                COMCIDP     911    A
COMCIDP                                                                                COMCIDP     912    A
COMCIDP   MX=TET   CON    "'?IDCTET"                                                   COMCIDP     913    A
COMCIDP   POT      SPACE  4,10                                                         COMCIDP     914    A
COMCIDP  **        POT - PARSING OPERATOR/OPERAND TABLE.                               COMCIDP     915    A
COMCIDP  *                                                                             COMCIDP     916    A
COMCIDP  *                                                                             COMCIDP     917    A
COMCIDP  *         *POT* IS THE PARSERS OPERATOR/OPERAND TABLE WHICH DETERMINES        COMCIDP     918    A
COMCIDP  *         HOW/WHEN/WHERE OPERATORS AND OPERANDS ARE PARSED.                   COMCIDP     919    A
COMCIDP  *                                                                             COMCIDP     920    A
COMCIDP  *         FIELDS WITHIN *POT* ARE DEFINED VIA SYMBOLS OF THE FORM --          COMCIDP     921    A
COMCIDP  *                                                                             COMCIDP     922    A
COMCIDP  *               PS.XXXP  AND  PS.XXXL                                         COMCIDP     923    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *                                                                  COMCIDP     924    A
COMCIDP  *         WHERE *PS.* IS A COMMON PREFIX, *XXX* IS A UNIQUE FIELD NAME,  COMCIDP     925    A
COMCIDP  *         AND *L/P* ARE THE *COMADEF* LENGTH AND RIGHT-MOST BIT POSITION  COMCIDP     926    A
COMCIDP  *         INDICATORS (SEE *COMADEF* FOR MORE INFO).                COMCIDP     927    A
COMCIDP  *                                                                  COMCIDP     928    A
COMCIDP  *         FIELDS, *XXX*, ARE --                                    COMCIDP     929    A
COMCIDP  *                                                                  COMCIDP     930    A
COMCIDP  *           PIP  = POLISH IN PROCESSOR ADDR.  *PIP* CONTAINS THE   COMCIDP     931    A
COMCIDP  *                  ADDR OF THE POLISH IN PROCESSOR FOR THIS OPERATOR/  COMCIDP     932    A
COMCIDP  *                  OPERAND.                                        COMCIDP     933    A
COMCIDP  *                                                                  COMCIDP     934    A
COMCIDP  *                  THIS FIELD HAS SIGNIFICANCE WHEN A PARTICULAR *RPN*  COMCIDP     935    A
COMCIDP  *                  HAS BEEN CONSTRUCTED **FOR** AN OPERATOR OR OPERAND  COMCIDP     936    A
COMCIDP  *                  **FROM** ITS CORRESPONDING *POT* ENTRY.  SEE BELOW.  COMCIDP     937    A
COMCIDP  *                                                                  COMCIDP     938    A
COMCIDP  *           POP  = POLISH OUT PROCESSOR ADDR.  *POP* CONTAINS THE  COMCIDP     939    A
COMCIDP  *                  ADDR OF THE POLISH OUT PROCESSOR FOR THIS OPERATOR/  COMCIDP     940    A
COMCIDP  *                  OPERAND.                                        COMCIDP     941    A
COMCIDP  *                                                                  COMCIDP     942    A
COMCIDP  *                  *POP* IS USED IN THE **CREATION** OF AN *RPN* ENTRY.  COMCIDP     943    A
COMCIDP  *                                                                  COMCIDP     944    A
COMCIDP  *           PRI  = PARSING PRIORITY FOR THIS OPERATOR (IGNORED FOR  COMCIDP     945    A
COMCIDP  *                  OPERANDS AND UNARY OPERATORS).                  COMCIDP     946    A
COMCIDP  *                                                                  COMCIDP     947    A
COMCIDP  *                  *PRI* HELPS DETERMINE THE ORDER IN WHICH OPERATORS  COMCIDP     948    A
COMCIDP  *                  GET POPPED OFF THE PARSING STACK (*PAST*), AND  COMCIDP     949    A
COMCIDP  *                  THEREFORE, THE ORDER IN WHICH THINGS COME OUT ON  COMCIDP     950    A
COMCIDP  *                  *RPN*.                                          COMCIDP     951    A
COMCIDP  *                                                                  COMCIDP     952    A
COMCIDP  *           UPIP = UNARY OPERATOR POLISH IN PROCESSOR ADDRESS.  *UPIP*  COMCIDP     953    A
COMCIDP  *                  IS USED WHEN AN OPERATOR HAS BOTH A BINARY **AND**  COMCIDP     954    A
COMCIDP  *                  A UNARY FORM, AS DOES MINUS  --                 COMCIDP     955    A
COMCIDP  *                                                                  COMCIDP     956    A
COMCIDP  *                       SNAP A-B                                   COMCIDP     957    A
COMCIDP  *                       SNAP -A                                    COMCIDP     958    A
COMCIDP  *                                                                  COMCIDP     959    A
COMCIDP  *         SEE ALSO *RPN* (REVERSE POLISH NOTATION TABLE) AND *PAS*  COMCIDP     960    A
COMCIDP  *         (PARSE SUBEXPRESSION).                                   COMCIDP     961    A
COMCIDP  *                                                                  COMCIDP     962    A
COMCIDP  *         NOTE THAT BECAUSE *PAS* USES TOKEN TYPES (*O.* SYMBOLS)  COMCIDP     963    A
COMCIDP  *         AS ORDINALS INTO *POT*, THAT THIS IMPLIES THAT *POT* ENTRIES  COMCIDP     964    A
COMCIDP  *         MUST BE IN THE SAME ORDER AS THE TOKEN TYPES.  THAT IS,  COMCIDP     965    A
COMCIDP  *         THERE IS A ONE-TO-ONE CORRESPONDENCE BETWEEN *O.* TOKEN  COMCIDP     966    A
COMCIDP  *         TYPES AND *POT*.  SEE ALSO *DEFTOT* (DEFINE TOKEN TYPE)  COMCIDP     967    A
COMCIDP  *         MACRO IN *COMSIDP* (IDP INTERFACE TEXT).                 COMCIDP     968    A
COMCIDP                                                                     COMCIDP     969    A
COMCIDP                                                                     COMCIDP     970    A
COMCIDP   '?IDSTOT SET    O.VAR-1                                           COMCIDP     971    A
COMCIDP                                                                     COMCIDP     972    A
COMCIDP   FW.POT   BSS    0           ** FWA **                             COMCIDP     973    A
COMCIDP   VAR      PASOP  POP=PAS24,PIP=PAS34                               COMCIDP     974    A
COMCIDP   CONS     PASOP  POP=PAS26,PIP=PAS34                               COMCIDP     975    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP   REG      PASOP  POP=PAS28,PIP=PAS34                                  COMCIDP     976    A
COMCIDP                                                                         COMCIDP     977    A
COMCIDP   PLUS     PASOP  POP=PAS20,PIP=PAS36,UPIP=PAS38,PRI=40B               COMCIDP     978    A
COMCIDP   MIN      PASOP  POP=PAS20,PIP=PAS40,UPIP=PAS42,PRI=40B               COMCIDP     979    A
COMCIDP   STAR     PASOP  POP=PAS20,PIP=PAS44,UPIP=PAS46,PRI=42B               COMCIDP     980    A
COMCIDP   LP       PASOP  POP=PAS16                                            COMCIDP     981    A
COMCIDP   RP       PASOP  POP=PAS18                                            COMCIDP     982    A
COMCIDP   SHFT     PASOP  POP=PAS20,PIP=PAS48,PRI=40B                          COMCIDP     983    A
COMCIDP   AND      PASOP  POP=PAS20,PIP=PAS52,PRI=40B                          COMCIDP     984    A
COMCIDP   OR       PASOP  POP=PAS20,PIP=PAS50,PRI=40B                          COMCIDP     985    A
COMCIDP   XOR      PASOP  POP=PAS20,PIP=PAS54,PRI=40B                          COMCIDP     986    A
COMCIDP   NOT      PASOP  POP=PAS20,UPIP=PAS42,PRI=77B                         COMCIDP     987    A
COMCIDP                                                                         COMCIDP     988    A
COMCIDP                                                                         COMCIDP     989    A
COMCIDP   L.POT    =      *-FW.POT                                             COMCIDP     990    A
COMCIDP   PAST     SPACE  4,10                                                 COMCIDP     991    A
COMCIDP  **        PAST - PARSING STACK.                                       COMCIDP     992    A
COMCIDP  *                                                                     COMCIDP     993    A
COMCIDP  *                                                                     COMCIDP     994    A
COMCIDP  *         *PAST* IS THE STACK USED BY THE PARSER *PAS* (PARSE         COMCIDP     995    A
COMCIDP  *         SUBEXPRESSION) DURING THE GENERATION AND EVALUATION         COMCIDP     996    A
COMCIDP  *         OF THE *RPN* (REVERSE POLISH NOTATION) TABLE.               COMCIDP     997    A
COMCIDP  *                                                                     COMCIDP     998    A
COMCIDP  *         DURING GENERATION OF *RPN* (I.E. POLISH OUT), *PAS*         COMCIDP     999    A
COMCIDP  *         PUSHS AND POPS OPERATORS ON AND OFF *PAST* IN ORDER TO GET  COMCIDP    1000    A
COMCIDP  *         THEM TO COME OUT IN THEIR PROPER ORDER ONTO *RPN*.          COMCIDP    1001    A
COMCIDP  *                                                                     COMCIDP    1002    A
COMCIDP  *         DURING THE EVALUATION OF *RPN* (I.E. POLISH IN), *PAS*      COMCIDP    1003    A
COMCIDP  *         PUSHS AND POPS OPERANDS ON AND OFF *PAST* IN ORDER TO GET   COMCIDP    1004    A
COMCIDP  *         THEM EVALUATED PROPERLY.  DURING THIS PHASE, THE FINAL      COMCIDP    1005    A
COMCIDP  *         ANSWER/EVALUATION IS THE ONLY THING LEFT ON *PAST* WHEN     COMCIDP    1006    A
COMCIDP  *         WE ARE DONE (ASSUMING NO ERRORS, OF COURSE).                COMCIDP    1007    A
COMCIDP  *                                                                     COMCIDP    1008    A
COMCIDP  *         SEE *PAS* AND *RPN*                                         COMCIDP    1009    A
COMCIDP                                                                         COMCIDP    1010    A
COMCIDP                                                                         COMCIDP    1011    A
COMCIDP   L.PAST   =      30                                                   COMCIDP    1012    A
COMCIDP            BSSZ   1                                                    COMCIDP    1013    A
COMCIDP   FW.PAST  BSSZ   L.PAST                                               COMCIDP    1014    A
COMCIDP   RPN      SPACE  4,10                                                 COMCIDP    1015    A
COMCIDP  **        RPN - REVERSE POLISH NOTATION TABLE.                        COMCIDP    1016    A
COMCIDP  *                                                                     COMCIDP    1017    A
COMCIDP  *                                                                     COMCIDP    1018    A
COMCIDP  *         *RPN* CONTAINS THE REVERSE POLISH NOTATION FORM OF EACH     COMCIDP    1019    A
COMCIDP  *         SUBEXPRESSION DURING PARSING BY *PAS* (PARSE SUBEXPRESSION). COMCIDP    1020    A
COMCIDP  *                                                                     COMCIDP    1021    A
COMCIDP  *         *RPN* IS GENERATED BY THE POLISH OUT PROCESSORS OF *PAS*,   COMCIDP    1022    A
COMCIDP  *         AND IS READ BY THE POLISH IN PROCESSORS OF *PAS*.  IT IS,   COMCIDP    1023    A
COMCIDP  *         THEREFORE, AN INTERNAL FORM OF COMMUNICATION WITHIN THE     COMCIDP    1024    A
COMCIDP  *         PARSER.                                                     COMCIDP    1025    A
COMCIDP  *                                                                     COMCIDP    1026    A
COMCIDP  *         INDIVIDUAL *RPN* ENTRIES ARE CONSTRUCTED FROM *POT* (PARSING COMCIDP    1027    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *        OPERATOR/OPERAND TABLE) AND *TB* (TOKEN BUFFER) BY THE          COMCIDP    1028    A
COMCIDP  *        POLISH OUT PROCESSORS.                                          COMCIDP    1029    A
COMCIDP  *                                                                        COMCIDP    1030    A
COMCIDP  *        *RPN* ENTRIES TAKE 2 GENERAL FORMS: OPERATOR ENTRIES, AND       COMCIDP    1031    A
COMCIDP  *        OPERAND ENTRIES.  BOTH FORMS ARE SIMILAR IN THAT THEY           COMCIDP    1032    A
COMCIDP  *        CONTAIN *POLISH IN* INFORMATION (FOR EXAMPLE, THE POLISH        COMCIDP    1033    A
COMCIDP  *        IN PROCESSOR ADDR FOR A PARTICULAR TOKEN/*RPN* ENTRY).          COMCIDP    1034    A
COMCIDP  *                                                                        COMCIDP    1035    A
COMCIDP  *        THE PRIMARY DIFFERENCE BETWEEN THESE 2 FORMS IS THAT OPERAND    COMCIDP    1036    A
COMCIDP  *        ENTRIES ALSO CONTAIN THE BINARY VALUE THAT *PAS*S POLISH OUT    COMCIDP    1037    A
COMCIDP  *        PROCESSORS WERE ABLE TO ASSOCIATE/EVALUATE WITH THIS OPERAND.   COMCIDP    1038    A
COMCIDP  *                                                                        COMCIDP    1039    A
COMCIDP  *        WHILE OPERATOR ENTRIES ARE 1 WORD IN LENGTH EACH, OPERAND       COMCIDP    1040    A
COMCIDP  *        ENTRIES ARE 2 WORDS EACH: THE EXTRA WORD CONTAINING THE         COMCIDP    1041    A
COMCIDP  *        BINARY VALUE OF THE OPERAND.                                    COMCIDP    1042    A
COMCIDP  *                                                                        COMCIDP    1043    A
COMCIDP  *        SEE ALSO *PAS*, *POT*.                                          COMCIDP    1044    A
COMCIDP  *                                                                        COMCIDP    1045    A
COMCIDP  *        REVERSE POLISH NOTATION EXAMPLES --                             COMCIDP    1046    A
COMCIDP  *                                                                        COMCIDP    1047    A
COMCIDP  *              A + B          A B +                                      COMCIDP    1048    A
COMCIDP  *              -A             A -                                        COMCIDP    1049    A
COMCIDP  *              *(A+B)         A B + *                                    COMCIDP    1050    A
COMCIDP  *              *(A$B)+C       A B $ * C +                                COMCIDP    1051    A
COMCIDP                                                                           COMCIDP    1052    A
COMCIDP                                                                           COMCIDP    1053    A
COMCIDP   L.RPN    =      L.PAST+L.PAST/2                                         COMCIDP    1054    A
COMCIDP   FW.RPN   BSSZ   L.RPN                                                   COMCIDP    1055    A
COMCIDP   .IDPOS   ENDIF                                                          CCGA018       8    A
COMCIDP   SNAPLNE  SPACE  4,10                                                    COMCIDP    1056    A
COMCIDP  **        SNAPLNE - OUTPUT LINE IMAGE AREA.                              COMCIDP    1057    A
COMCIDP  *                                                                        COMCIDP    1058    A
COMCIDP  *                                                                        COMCIDP    1059    A
COMCIDP                                                                           COMCIDP    1060    A
COMCIDP                                                                           COMCIDP    1061    A
COMCIDP   SNAPLNE  BSSZ   15          INTERACTIVE AND BATCH DEBUG OUTPUT LINE     COMCIDP    1062    A
COMCIDP  *                              IMAGE AREA                                COMCIDP    1063    A
COMCIDP   TB=FLL   SPACE  4,10                                                    COMCIDP    1064    A
COMCIDP   .IDPOS   IFNE   .OS,2                                                   CCGA018       9    A
COMCIDP  **        TB=FLL - CURRENT *FWA,LWA,LEN* KEYWORD POINTER.                COMCIDP    1065    A
COMCIDP  *                                                                        COMCIDP    1066    A
COMCIDP  *                                                                        COMCIDP    1067    A
COMCIDP  *        *TB=FLL* CONTAINS THE ADDR OF THE MOST RECENT *KEYW*            COMCIDP    1068    A
COMCIDP  *        ENTRY FOR A *PTR* DRIVEN KEYWORD.  THAT IS, A KEYWORD           COMCIDP    1069    A
COMCIDP  *        WHICH HAS THE *PTR* FIELD IN ITS *KEYW* ENTRY .NZ.              COMCIDP    1070    A
COMCIDP  *                                                                        COMCIDP    1071    A
COMCIDP  *        (TB=FLL) ARE USED BY THE *UP* AND *DOWN* PROCESSORS             COMCIDP    1072    A
COMCIDP  *        FOR DETERMINING WHICH *KEYW* POINTER (AT *FW.PTR* ET SEQ)       COMCIDP    1073    A
COMCIDP  *        THE INTERACTIVE USER WISHES TO MOVE *UP* OR *DOWN*.             COMCIDP    1074    A
COMCIDP                                                                           COMCIDP    1075    A
COMCIDP                                                                           COMCIDP    1076    A
COMCIDP   TB=FLL   BSSZ   1                                                       COMCIDP    1077    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP   TB=KEY   SPACE  4,10                                          COMCIDP     1078     A
COMCIDP  **        TB=KEY - CURRENT KEYWORD POINTER.                   COMCIDP     1079     A
COMCIDP  *                                                             COMCIDP     1080     A
COMCIDP  *                                                             COMCIDP     1081     A
COMCIDP  *         *TB=KEY* CONTAINS THE ADDRESS OF THE MOST RECENT *KEYW*  COMCIDP  1082  A
COMCIDP  *         ENTRY FOUND BY *CST* (CLASSIFY STMT).  IT IS USED   COMCIDP     1083     A
COMCIDP  *         WHENEVER ANYONE WISHES TO GET TO ANY INFORMATION CONTAINED  COMCIDP  1084  A
COMCIDP  *         IN THE *KEYW* ENTRY FOR THE CURRENT STMT.           COMCIDP     1085     A
COMCIDP  *                                                             COMCIDP     1086     A
COMCIDP  *         FOR EXAMPLE, *IDP*S MAIN LOOP USES (TB=KEY) TO DETERMINE  COMCIDP  1087  A
COMCIDP  *         WHICH STMT PROCESSOR TO TRANSFER CONTROL TO FOR A STMT  COMCIDP  1088  A
COMCIDP  *         IN THE TOKEN BUFFER.  THAT IS, AFTER A *CST* CALL, (TB=KEY)  COMCIDP  1089  A
COMCIDP  *         POINTS TO THE *FOUND* KEYWORD ENTRY.                COMCIDP     1090     A
COMCIDP                                                                COMCIDP     1091     A
COMCIDP                                                                COMCIDP     1092     A
COMCIDP   TB=KEY   BSSZ   1                                            COMCIDP     1093     A
COMCIDP   TB=STMT  SPACE  4,10                                         COMCIDP     1094     A
COMCIDP  **        TB=STMT - CURRENT STMT POINTER.                     COMCIDP     1095     A
COMCIDP  *                                                             COMCIDP     1096     A
COMCIDP  *                                                             COMCIDP     1097     A
COMCIDP  *         *TB=STMT* CONTAINS INFORMATION ABOUT THE CURRENT STMT  COMCIDP  1098  A
COMCIDP  *         IN THE TOKEN BUFFER, *IDPTB*.  *TB=STMT* IS USED IN 2  COMCIDP  1099  A
COMCIDP  *         DISTINCTLY DIFFERENT WAYS --                        COMCIDP     1100     A
COMCIDP  *                                                             COMCIDP     1101     A
COMCIDP  *           1. DURING TOKEN GENERATION OF A SOURCE LINE BY *COMCTOK*,  COMCIDP  1102  A
COMCIDP  *              *TB=STMT* CONTAINS THE ADDR OF THE STMT INFORMATION  COMCIDP  1103  A
COMCIDP  *              HEADER WORD FOR THE STMT BEING ENTOKENED.      COMCIDP     1104     A
COMCIDP  *                                                             COMCIDP     1105     A
COMCIDP  *              WHEN *TOK* ENCOUNTERS AN END-OF-STMT CONDITION (I.E.  COMCIDP  1106  A
COMCIDP  *              EITHER END-OF-LINE OR * *), THEN *TOK=EOS* (END-OF-  COMCIDP  1107  A
COMCIDP  *              STMT PROCESSING FOR *COMCTOK*) USES (TB=STMT) TO FIND  COMCIDP  1108  A
COMCIDP  *              THE STMT INFO HEADER WORD FOR THE STMT JUST ENTOKENED  COMCIDP  1109  A
COMCIDP  *              SO THAT HE CAN GO BACK AND FILL IN THE LENGTH OF THIS  COMCIDP  1110  A
COMCIDP  *              STMT.                                          COMCIDP     1111     A
COMCIDP  *                                                             COMCIDP     1112     A
COMCIDP  *           2. DURING THE STMT PROCESSING PHASE, *TB=STMT* CONTAINS  COMCIDP  1113  A
COMCIDP  *              THE ADDR OF THE CURRENT STMT BEING PROCESSED.  IN THIS  COMCIDP  1114  A
COMCIDP  *              WAY, IDP.S MAIN LOOP CAN ADVANCE TO EACH NEW STMT THAT  COMCIDP  1115  A
COMCIDP  *              OCCURS ON A SINGLE SOURCE LINE.                COMCIDP     1116     A
COMCIDP                                                                COMCIDP     1117     A
COMCIDP                                                                COMCIDP     1118     A
COMCIDP   TB=STMT  CON    IDPTB                                        COMCIDP     1119     A
COMCIDP   DATA     SPACE  4,10                                         COMCIDP     1120     A
COMCIDP  **        *IDP* DATA DECLARATIONS.                            COMCIDP     1121     A
COMCIDP                                                                COMCIDP     1122     A
COMCIDP                                                                COMCIDP     1123     A
COMCIDP   IDPSTO   BSSZ   3                                            COMCIDP     1124     A
COMCIDP                                                                COMCIDP     1125     A
COMCIDP            LOC    IDPSTO                                       COMCIDP     1126     A
COMCIDP   IDPSTA   =      *          ADDR TO STORE INTO FOR *STORE* PROCESSING  COMCIDP  1127  A
COMCIDP   IDPSTC   =      *+1        NEW VALUE ASSEMBLED FROM C1 THRU C4 ON  COMCIDP  1128  A
COMCIDP  *                             *STORE* COMMAND                 COMCIDP     1129     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP   IDPSTM   =      *+2         MASK ASSEMBLED FROM NULL C1 THRU C4 ON          COMCIDP    1130    A
COMCIDP   *                              *STORE* COMMAND. IDP USES THIS MASK TO        COMCIDP    1131    A
COMCIDP   *                              PICK UP THE PREVIOUS CONTENTS OF THE C-N-     COMCIDP    1132    A
COMCIDP   *                              FIELD IF THE NEW C-N- FIELD IS NULL           COMCIDP    1133    A
COMCIDP   *                              E.G.  STORE ADDR,,22222,,44444                COMCIDP    1134    A
COMCIDP   *                              PREVIOUS (STA) = 01020304050607080910         COMCIDP    1135    A
COMCIDP   *                                       (STC) = 00000222220000044444         COMCIDP    1136    A
COMCIDP   *                                       (STM) = 77777000007777700000         COMCIDP    1137    A
COMCIDP   *                                   NEW (STA) = 01020222220607044444         COMCIDP    1138    A
COMCIDP            LOC    *O                                                           COMCIDP    1139    A
COMCIDP   IDPBTC   BSSZ   1           BREAK TYPE CODE                                  COMCIDP    1140    A
COMCIDP                                                                                COMCIDP    1141    A
COMCIDP   IDPACC   BSSZ   1           ACCUMULATOR FOR THIS SUBEXPRESSION               COMCIDP    1142    A
COMCIDP   IDPNUL   BSSZ   1           .ZR. IF JUST PARSED NULL EXPRESSION              COMCIDP    1143    A
COMCIDP   *                                E.G.   SNAP FWA,,LEN   (LWA IS NULL)        COMCIDP    1144    A
COMCIDP                                                                                COMCIDP    1145    A
COMCIDP   IDP0TR   BSSZ   1           REGISTER DESIGNATOR                              COMCIDP    1146    A
COMCIDP   *                              USED FOR *REG R1,R2,...,RN* PROCESSING.       COMCIDP    1147    A
COMCIDP   *                              FORMAT IS--  60/0TR WHERE                     COMCIDP    1148    A
COMCIDP   *                                  T = REGISTER TYPE (B=0,A=1,X=2)           COMCIDP    1149    A
COMCIDP   *                                  R = REGISTER NR (0-7)                     COMCIDP    1150    A
COMCIDP                                                                                COMCIDP    1151    A
COMCIDP   IDPNAM   BSSZ   1           SAVE CELL FOR *ST=SET* -- SET NAME (0L FMT)      COMCIDP    1152    A
COMCIDP                                                                                COMCIDP    1153    A
COMCIDP   IDPFAD   BSSZ   1           .NZ. IF *DAZ* IS TO FORCE OUT ADDR OF INST-      COMCIDP    1154    A
COMCIDP   *                              RUCTION BEING DEASSEMBLED EVEN IF NOT         COMCIDP    1155    A
COMCIDP   *                              FORCED UPPER IN WORD, ELSE .ZR.               COMCIDP    1156    A
COMCIDP   *                              E.G. IF *FAD* IS .ZR. --                      COMCIDP    1157    A
COMCIDP   *                              ADDR  54111                SA1  A1+B1         COMCIDP    1158    A
COMCIDP   *                                        0311000000       NZ   X1,...        COMCIDP    1159    A
COMCIDP   *                                                  10611  BX6  X1            COMCIDP    1160    A
COMCIDP   *                              E.G. IF *FAD* IS .NZ. --                      COMCIDP    1161    A
COMCIDP   *                              ADDR  54111                SA1  A1+B1         COMCIDP    1162    A
COMCIDP   *                              ADDR       0311000000      NZ   X1,...        COMCIDP    1163    A
COMCIDP   *                              ADDR                 10611  BX6  X1            COMCIDP    1164    A
COMCIDP   *                              NOTE-- *DAZ* WILL ALWAYS ZERO *FAD*           COMCIDP    1165    A
COMCIDP                                                                                COMCIDP    1166    A
COMCIDP   IDPPPP   BSSZ   2           PSEUDO P REGISTER AND POS COUNTER                COMCIDP    1167    A
COMCIDP                                                                                COMCIDP    1168    A
COMCIDP            LOC    IDPPPP                                                       COMCIDP    1169    A
COMCIDP   IDPPREG  =      *           PSEUDO P REGISTER. POINTS TO WORD CONTAIN-       COMCIDP    1170    A
COMCIDP   *                              ING NEXT INSTRUCTION TO BE XEQ                COMCIDP    1171    A
COMCIDP   IDPPOS   =      *+1         POSITION COUNTER. LEFT MOST BIT OF INST-         COMCIDP    1172    A
COMCIDP   *                              RUCTION TO XEQ IS BIT (IDPPOS)-1 IN           COMCIDP    1173    A
COMCIDP   *                              ((IDPPREG)).(BITS ARE NUMBERED 59 THRU 0)     COMCIDP    1174    A
COMCIDP            LOC    *O                                                           COMCIDP    1175    A
COMCIDP                                                                                COMCIDP    1176    A
COMCIDP   IDPXAR   BSSZ   1           .MI. IF NO ADDR REF VIA LOAD/STORE, ELSE         COMCIDP    1177    A
COMCIDP   *                              ADDR THAT WAS REFERENCED. USED IN *STAR*      COMCIDP    1178    A
COMCIDP   *                              PROCESSING.                                   COMCIDP    1179    A
COMCIDP   IDPXCA   BSSZ   2           CONDITIONAL *STEP* COMPARE ADDRESSES.            COMCIDP    1180    A
COMCIDP   *                              IF THIS IS A CONDITIONAL STEP (STNE,...),     COMCIDP    1181    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *                          THEN ((XA1)) WILL BE COMPARED TO ((XA2))        COMCIDP    1182    A
COMCIDP                                                                             COMCIDP    1183    A
COMCIDP            LOC    IDPXCA                                                    COMCIDP    1184    A
COMCIDP   IDPXA1   =      *                                                         COMCIDP    1185    A
COMCIDP   IDPXA2   =      *+1                                                       COMCIDP    1186    A
COMCIDP            LOC    *O                                                        COMCIDP    1187    A
COMCIDP                                                                             COMCIDP    1188    A
COMCIDP   IDPXFR   BSSZ   1           .MI. IF NO XFER OF CONTROL BY BRANCH OR RJ,   COMCIDP    1189    A
COMCIDP  *                            ELSE ADDR TO XFER CONTROL TO. USED IN         COMCIDP    1190    A
COMCIDP  *                            STEP MODE SO THAT PSEUDO P REGISTER IS        COMCIDP    1191    A
COMCIDP  *                            NOT PREMATURELY UPDATED WHEN BRANCHING.       COMCIDP    1192    A
COMCIDP   IDPXLST  BSSZ   1           .NZ. IF LISTING THIS INSTRUCTION IN STEP      COMCIDP    1193    A
COMCIDP  *                            MODE, ELSE .ZR.                               COMCIDP    1194    A
COMCIDP   IDPXRJ   BSSZ   1           .ZR. IF AT LEVEL 0 (I.E. NO *RJ* ACTIVE),     COMCIDP    1195    A
COMCIDP  *                            ELSE ADDR OF ENTRY POINT TO ROUTINE           COMCIDP    1196    A
COMCIDP  *                            CALLED FROM LEVEL 0. USED IN STEP MODE TO     COMCIDP    1197    A
COMCIDP  *                            DETERMINE WHEN TO TURN LISTING BACK ON        COMCIDP    1198    A
COMCIDP  *                            WHEN /-RJ/ OPTION SELECTED                    COMCIDP    1199    A
COMCIDP   IDPXTC   BSSZ   1           *STEP* TYPE CODE                              COMCIDP    1200    A
COMCIDP                                                                             COMCIDP    1201    A
COMCIDP   IDPXOP   BSSZ   2           STEP OPTIONS FLAGS. THERE IS A ONE-TO-ONE     COMCIDP    1202    A
COMCIDP  *                            RELATIONSHIP BETWEEN THE STEP OPTIONS         COMCIDP    1203    A
COMCIDP  *                            FLAGS AND THE STEP OPTIONS KEYWORD TABLE.     COMCIDP    1204    A
COMCIDP  *                            WHEN *POL* RETURNS WITH A KEYWORD FIND        COMCIDP    1205    A
COMCIDP  *                            AND AN OPTIONS FLAG VALUE, THE ORDINAL OF     COMCIDP    1206    A
COMCIDP  *                            FIND IN THE STEP OPTIONS KEYWORD TABLE IS     COMCIDP    1207    A
COMCIDP  *                            THE ORDINAL TO STORE INTO *IDPXOP*.           COMCIDP    1208    A
COMCIDP                                                                             COMCIDP    1209    A
COMCIDP            LOC    IDPXOP                                                    COMCIDP    1210    A
COMCIDP   IDPXOL   =      *           .NZ. IF /L/ SELECTED ON STEP COMAND,          COMCIDP    1211    A
COMCIDP  *                            ELSE .ZR. IF /-L/ SELECTED                    COMCIDP    1212    A
COMCIDP   IDPXORJ  =      *+1         .NZ. IF /RJ/ SELECTED ON STEP COMMAND,        COMCIDP    1213    A
COMCIDP  *                            ELSE .ZR. IF /-RJ/ SELECTED                   COMCIDP    1214    A
COMCIDP            LOC    *O                                                        COMCIDP    1215    A
COMCIDP   MSG      SPACE  4,10                                                      COMCIDP    1216    A
COMCIDP  **        *IDP* MESSAGES.                                                  COMCIDP    1217    A
COMCIDP                                                                             COMCIDP    1218    A
COMCIDP   IDPMSG2  DIS    3, CALLD BY 000000 IN XXXXXXX                             COMCIDP    1219    A
COMCIDP   IDPMSG4  DIS    3, OLD VALU 00000000000000000000                          COMCIDP    1220    A
COMCIDP            DATA   0                                                         COMCIDP    1221    A
COMCIDP   IDPMSG5  DIS    3, NEW VALU 00000000000000000000                          COMCIDP    1222    A
COMCIDP            DATA   0                                                         COMCIDP    1223    A
COMCIDP   PATFLG   SPACE  4,10                                                      COMCIDP    1224    A
COMCIDP  **        PATFLG - PARSE *FWA,LWA,LEN* TRIPLE FLAG.                        COMCIDP    1225    A
COMCIDP  *                                                                          COMCIDP    1226    A
COMCIDP  *                                                                          COMCIDP    1227    A
COMCIDP  *         CONTAINS STATUS ABOUT THE *FWA,LWA,LEN* TRIPLE BEING             COMCIDP    1228    A
COMCIDP  *         PARSED/PROCESSED BY ROUTINE *PAT*.  USED SO THAT *PAT*           COMCIDP    1229    A
COMCIDP  *         CAN RESOLVE ANY POSSIBLE SYNTACTIC DIFFERENCES/PROBLEMS          COMCIDP    1230    A
COMCIDP  *         THAT MIGHT OCCUR BETWEEN THE *LWA* AND *LEN* PARAMETERS.         COMCIDP    1231    A
COMCIDP  *         SEE SUBROUTINE *PAT*.                                            COMCIDP    1232    A
COMCIDP  *                                                                          COMCIDP    1233    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *         BIT FLAGS WITHIN *PATFLG* MUST EXIST IN BITS 0 THRU 16          COMCIDP    1234    A
COMCIDP  *         (I.E. FIELD MANIPULATION IS DONE VIA *SX.I* TYPE INSTRUCTIONS)  COMCIDP    1235    A
COMCIDP  *         AND ARE DESCRIBE/DEFINED IN THE USUAL MANNER VIA SYMBOLS        COMCIDP    1236    A
COMCIDP  *         OF THE FORM --                                                  COMCIDP    1237    A
COMCIDP  *                                                                         COMCIDP    1238    A
COMCIDP  *               PAF.XXXP  AND  PAF.XXXL                                   COMCIDP    1239    A
COMCIDP  *                                                                         COMCIDP    1240    A
COMCIDP  *         WHERE *PAF.* IS A COMMON PREFIX AND *XXX* IS A UNIQUE           COMCIDP    1241    A
COMCIDP  *         BIT FLAG (FIELD) NAME.                                          COMCIDP    1242    A
COMCIDP  *                                                                         COMCIDP    1243    A
COMCIDP  *         BIT FLAGS *XXX* ARE --                                          COMCIDP    1244    A
COMCIDP  *                                                                         COMCIDP    1245    A
COMCIDP  *         LWA  = 1 IF LWA WAS SPECIFIED EXPLICITLY (I.E. NON-NULL),       COMCIDP    1246    A
COMCIDP  *                ELSE 0.                                                  COMCIDP    1247    A
COMCIDP  *         LEN  = 1 IF LEN WAS SPECIFIED EXPLICITLY (I.E. NON-NULL),       COMCIDP    1248    A
COMCIDP  *                ELSE 0.                                                  COMCIDP    1249    A
COMCIDP                                                                            COMCIDP    1250    A
COMCIDP                                                                            COMCIDP    1251    A
COMCIDP   PATFLG   BSSZ   1                                                        COMCIDP    1252    A
COMCIDP   PTR      SPACE  4,10                                                     COMCIDP    1253    A
COMCIDP  **        PTR - POINTER AREA.                                             COMCIDP    1254    A
COMCIDP  *                                                                         COMCIDP    1255    A
COMCIDP  *                                                                         COMCIDP    1256    A
COMCIDP                                                                            COMCIDP    1257    A
COMCIDP                                                                            COMCIDP    1258    A
COMCIDP   N.PTR    =      10          NR OF *PTR* ENTRIES                          COMCIDP    1259    A
COMCIDP                                                                            COMCIDP    1260    A
COMCIDP   FW.PTR   BSS    0                                                        COMCIDP    1261    A
COMCIDP   '?ID#001 DUP    N.PTR                                                    COMCIDP    1262    A
COMCIDP            DUP    L1.PTR,1                                                 COMCIDP    1263    A
COMCIDP            VFD    PT.FWAL/0,PT.LENL/1                                      COMCIDP    1264    A
COMCIDP   '?ID#001 ENDD                                                            COMCIDP    1265    A
COMCIDP                                                                            COMCIDP    1266    A
COMCIDP                                                                            COMCIDP    1267    A
COMCIDP   L.PTR    =      *-FW.PTR                                                 COMCIDP    1268    A
COMCIDP   TC=      SPACE  4,10                                                     COMCIDP    1269    A
COMCIDP  **        TC= - *TOKCOM*, THE *USER/TOK* COMMUNICATIONS AREA.             COMCIDP    1270    A
COMCIDP  *                                                                         COMCIDP    1271    A
COMCIDP  *                                                                         COMCIDP    1272    A
COMCIDP  *         SYMBOLS WITH THE COMMON PREFIX *TC=* CONTAIN INFORMATION        COMCIDP    1273    A
COMCIDP  *         THAT IS TO BE PASSED BETWEEN IDP AND *COMCTOK* (THE COMMON      COMCIDP    1274    A
COMCIDP  *         TOKEN GENERATOR).                                               COMCIDP    1275    A
COMCIDP  *                                                                         COMCIDP    1276    A
COMCIDP  *         LOCATIONS WITHIN *TOKCOM* ARE DEFINED VIA ORDINALS.  THESE      COMCIDP    1277    A
COMCIDP  *         ORDINALS ARE DEFINED VIA SYMBOLS OF THE FORM --                 COMCIDP    1278    A
COMCIDP  *                                                                         COMCIDP    1279    A
COMCIDP  *               TC.XXX                                                    COMCIDP    1280    A
COMCIDP  *                                                                         COMCIDP    1281    A
COMCIDP  *         WHERE *TC.* IS A COMMON PREFIX, AND *XXX* IS A *TOKCOM*         COMCIDP    1282    A
COMCIDP  *         LOCATION ORDINAL NAME (*TC.XXX* SYMBOLS ARE DEFINED             COMCIDP    1283    A
COMCIDP  *         IN *COMSTOK*, THE COMMON TOKEN GENERATOR INTERFACE TEXT).       COMCIDP    1284    A
COMCIDP  *                                                                         COMCIDP    1285    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *         TO AVOID BUGS, MISUNDERSTANDINGS, AND GENERAL CONFUSION,          COMCIDP    1286    A
COMCIDP  *         **ALL** LOCATIONS WITHIN *TOKCOM* SHOULD BE DEFINED/REFERENCED    COMCIDP    1287    A
COMCIDP  *         VIA *TC.XXX* SYMBOLS.                                             COMCIDP    1288    A
COMCIDP                                                                              COMCIDP    1289    A
COMCIDP                                                                              COMCIDP    1290    A
COMCIDP   FW.TCOM  BSSZ   L.TCOM                                                     COMCIDP    1291    A
COMCIDP                                                                              COMCIDP    1292    A
COMCIDP            LOC    FW.TCOM                                                    COMCIDP    1293    A
COMCIDP   TC=P     =      *+TC.P                                                     COMCIDP    1294    A
COMCIDP   TC=SOA   =      *+TC.SOA                                                   COMCIDP    1295    A
COMCIDP   TC=SOX   =      *+TC.SOX                                                   COMCIDP    1296    A
COMCIDP   TC=SOL   =      *+TC.SOL                                                   COMCIDP    1297    A
COMCIDP   TC=SOB   =      *+TC.SOB                                                   COMCIDP    1298    A
COMCIDP   TC=SOC   =      *+TC.SOC                                                   COMCIDP    1299    A
COMCIDP   TC=TBA   =      *+TC.TBA                                                   COMCIDP    1300    A
COMCIDP   TC=TBB   =      *+TC.TBB                                                   COMCIDP    1301    A
COMCIDP   TC=RSF   =      *+TC.RSF                                                   COMCIDP    1302    A
COMCIDP   TC=BOL   =      *+TC.BOL                                                   COMCIDP    1303    A
COMCIDP   TC=EOL   =      *+TC.EOL                                                   COMCIDP    1304    A
COMCIDP   TC=REST  =      *+TC.REST                                                  COMCIDP    1305    A
COMCIDP            LOC    *O                                                         COMCIDP    1306    A
COMCIDP   IDPTB    SPACE  4,10                                                       COMCIDP    1307    A
COMCIDP  **        IDPTB - COMMAND LINE TOKEN BUFFER.                                COMCIDP    1308    A
COMCIDP  *                                                                           COMCIDP    1309    A
COMCIDP  *                CONTAINS THE CURRENT COMMAND LINE IN TOKEN FORM.           COMCIDP    1310    A
COMCIDP  *                                                                           COMCIDP    1311    A
COMCIDP  *         TOKEN BUFFER FORMAT --                                            COMCIDP    1312    A
COMCIDP  *                                                                           COMCIDP    1313    A
COMCIDP  *         VFD    1/LAS,41/0,18/LEN   (TOKEN INFO WORD FOR STMT 1)           COMCIDP    1314    A
COMCIDP  *         VFD    42/0LCHARS,18/TOKEN TYPE                                   COMCIDP    1315    A
COMCIDP  *         VFD    42/0LCHAR ,18/TOKEN TYPE                                   COMCIDP    1316    A
COMCIDP  *          .                                                                COMCIDP    1317    A
COMCIDP  *          .                                                                COMCIDP    1318    A
COMCIDP  *         VFD    60/0   (END OF STATEMENT MARK FOR STMT 1)                  COMCIDP    1319    A
COMCIDP  *                                                                           COMCIDP    1320    A
COMCIDP  *         VFD    1/LAS,41/0,18/LEN   (TOKEN INFO WORD FOR STMT 2)           COMCIDP    1321    A
COMCIDP  *          .                                                                COMCIDP    1322    A
COMCIDP  *         ETC                                                               COMCIDP    1323    A
COMCIDP  *                                                                           COMCIDP    1324    A
COMCIDP  *                THE FIELDS IN THE TOKEN INFORMATION WORDS ARE DESCRIBED    COMCIDP    1325    A
COMCIDP  *         BY PAIRS OF SYMBOLS OF THE FORM *TB.XXXP* AND *TB.XXXL*,          COMCIDP    1326    A
COMCIDP  *         WHERE *TB* IS THE COMMON PREFIX, AND *XXX* IS THE FIELD           COMCIDP    1327    A
COMCIDP  *         DESCRIPTION.                                                      COMCIDP    1328    A
COMCIDP  *                                                                           COMCIDP    1329    A
COMCIDP  *         FIELD DESCRIPTIONS (XXX) ARE --                                   COMCIDP    1330    A
COMCIDP  *                                                                           COMCIDP    1331    A
COMCIDP  *         LAS  = 1  IF THIS IS LAST STMT ON LINE, ELSE 0                    COMCIDP    1332    A
COMCIDP  *                                                                           COMCIDP    1333    A
COMCIDP  *         LEN  = NR OF WORDS OR TOKENS IN STMT  (INCLUDES EOL MARK)         COMCIDP    1334    A
COMCIDP  *                  NOTE--IF LAS .EQ. 0, TOKEN INFO WORD FOR NEXT            COMCIDP    1335    A
COMCIDP  *                  STMT IS AT -- TOKEN INFO WORD + LEN + 1                  COMCIDP    1336    A
COMCIDP                                                                              COMCIDP    1337    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP                                                                                COMCIDP     1338     A
COMCIDP   L.TB     =      100         LENGTH OF TOKEN BUFFER                           COMCIDP     1339     A
COMCIDP   IDPTB    BSSZ   L.TB                                                         COMCIDP     1340     A
COMCIDP   SCT      SPACE  4,10                                                         COMCIDP     1341     A
COMCIDP  **        SCT - STATEMENT CONTROL TOKEN TABLE.                                COMCIDP     1342     A
COMCIDP  *                                                                             COMCIDP     1343     A
COMCIDP  *                                                                             COMCIDP     1344     A
COMCIDP  *         THE FOLLOWING IS THE TABLE CONTAINING ALL THE *STMT                 COMCIDP     1345     A
COMCIDP  *         CONTROL* TOKENS AND THEIR RESPECTIVE PROCESSOR ADDRESSES            COMCIDP     1346     A
COMCIDP  *         FOR ALL THE SPECIAL *STMT CONTROL* TOKENS THAT CAN                  COMCIDP     1347     A
COMCIDP  *         PRECEDE AN IDP STMT KEYWORD.                                        COMCIDP     1348     A
COMCIDP                                                                                COMCIDP     1349     A
COMCIDP                                                                                COMCIDP     1350     A
COMCIDP   FW.SCT   BSS    0           ** FWA **                                        COMCIDP     1351     A
COMCIDP   PER      SCTOT  SC=BRF      .                                                COMCIDP     1352     A
COMCIDP   QQQ      SCTOT  SC=QQQ      ?                                                COMCIDP     1353     A
COMCIDP            DATA   0           ** END OF TABLE **                               COMCIDP     1354     A
COMCIDP   KEY      SPACE  4,10                                                         COMCIDP     1355     A
COMCIDP  **        *IDP* KEYWORD TABLE.                                                COMCIDP     1356     A
COMCIDP  *                                                                             COMCIDP     1357     A
COMCIDP  *                                                                             COMCIDP     1358     A
COMCIDP                                                                                COMCIDP     1359     A
COMCIDP                                                                                COMCIDP     1360     A
COMCIDP   FW=KEY   CON    FW.KEY                                                       COMCIDP     1361     A
COMCIDP   FW.KEY   SPACE  4,10                                                         COMCIDP     1362     A
COMCIDP   FW.KEY   BSS    0           ** FWA OF KEYW TABLE **                          COMCIDP     1363     A
COMCIDP   ABS      KEYW   ABS                                                          COMCIDP     1364     A
COMCIDP   COD      KEYW   (CODE FWA,LWA,LEN)                                           COMCIDP     1365     A
COMCIDP   CON      KEYW   CONNECT                                                      COMCIDP     1366     A
COMCIDP   DPC      KEYW   (DPC FWA,LWA,LEN)                                            COMCIDP     1367     A
COMCIDP   DSC      KEYW   DISCONT                                                      COMCIDP     1368     A
COMCIDP   END      KEYW   END                                                          COMCIDP     1369     A
COMCIDP   FRZ      KEYW   FREEZE                                                       COMCIDP     1370     A
COMCIDP   JP       KEYW   (JUMP ADDR)                                                  COMCIDP     1371     A
COMCIDP   OPT      KEYW   (OPTION P1,...,PN)                                           COMCIDP     1372     A
COMCIDP   OUT      KEYW   (OUTPUT I,B,IE,BE)                                           COMCIDP     1373     A
COMCIDP   REG      KEYW   (REGS R1,...,R.N)                                            COMCIDP     1374     A
COMCIDP   REL      KEYW   REL                                                          COMCIDP     1375     A
COMCIDP   SET      KEYW   (SET NAME,VALUE)                                             COMCIDP     1376     A
COMCIDP   SNP      KEYW   (SNAP FWA,LWA,LEN)                                           COMCIDP     1377     A
COMCIDP   STO      KEYW   (STORE ADDR,C1,C2,C3,C4)                                     COMCIDP     1378     A
COMCIDP   STA      KEYW   STATUS                                                       COMCIDP     1379     A
COMCIDP   TIM      KEYW   TIME                                                         COMCIDP     1380     A
COMCIDP   UBK      KEYW   (UNBREAK BRAD1,...,BRAD.N)                                   COMCIDP     1381     A
COMCIDP   UST      KEYW   (UNSET NAME1,...,NAME.N)                                     COMCIDP     1382     A
COMCIDP   UBK      KEYW   (UB BRAD1,...,BRAD.N)                                        COMCIDP     1383     A
COMCIDP   UST      KEYW   (US NAME1,...,NAME.N)                                        COMCIDP     1384     A
COMCIDP   WHR      KEYW   (WHERE ADDR1,...,ADDR.N)                                     COMCIDP     1385     A
COMCIDP   XEQ      KEYW   XEQ                                                          COMCIDP     1386     A
COMCIDP   XFR      KEYW   (XFER NR)                                                    COMCIDP     1387     A
COMCIDP                                                                                COMCIDP     1388     A
COMCIDP  *         *BREAK* KEYWORDS.                                                   COMCIDP     1389     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *                                                                  COMCIDP    1390    A
COMCIDP  *         NOTE THAT THE *BREAK* TYPE CODE (BTC) IS PLACED IN THE *XAT*  COMCIDP    1391    A
COMCIDP  *         FIELD IN WORD 1 OF EACH *BREAK* KEYWORD.                 COMCIDP    1392    A
COMCIDP  *                                                                  COMCIDP    1393    A
COMCIDP  *         *BTC* VALUES HAVE THE FOLLOWING SIGNIFICANCE --          COMCIDP    1394    A
COMCIDP  *                                                                  COMCIDP    1395    A
COMCIDP  *                BTC = 0  IS A BREAK WITH ONLY ONE ADDRESS EXPRESSION--  COMCIDP    1396    A
COMCIDP  *                            BREAK ADDR                            COMCIDP    1397    A
COMCIDP  *                BTC .LT. BTC.EQ  ARE BREAKS WITH 2 ADDR EXPRESSIONS--  COMCIDP    1398    A
COMCIDP  *                            BRPL ADDR,ADDR1                       COMCIDP    1399    A
COMCIDP  *                BTC .GE. BTC.EQ  ARE BREAKS WITH 3 ADDR EXPRESSIONS--  COMCIDP    1400    A
COMCIDP  *                            BREQ ADDR,ADDR1,ADDR2                 COMCIDP    1401    A
COMCIDP  *                                                                  COMCIDP    1402    A
COMCIDP  *           THIS INFORMATION IS USED THROUGHOUT BREAK PROCESSING.  COMCIDP    1403    A
COMCIDP                                                                     COMCIDP    1404    A
COMCIDP   BRK      KEYW   (BREAK BRAD,LL,UL,INC),0                          COMCIDP    1405    A
COMCIDP   BRK      KEYW   (BRPL BRAD,A1,LL,UL,INC),BTC.PL                   COMCIDP    1406    A
COMCIDP   BRK      KEYW   (BRMI BRAD,A1,LL,UL,INC),BTC.MI                   COMCIDP    1407    A
COMCIDP   BRK      KEYW   (BRZR BRAD,A1,LL,UL,INC),BTC.ZR                   COMCIDP    1408    A
COMCIDP   BRK      KEYW   (BRNZ BRAD,A1,LL,UL,INC),BTC.NZ                   COMCIDP    1409    A
COMCIDP   BRK      KEYW   (BREQ BRAD,A1,A2,LL,UL,INC),BTC.EQ                COMCIDP    1410    A
COMCIDP   BRK      KEYW   (BRNE BRAD,A1,A2,LL,UL,INC),BTC.NE                COMCIDP    1411    A
COMCIDP   BRK      KEYW   (BRLT BRAD,A1,A2,LL,UL,INC),BTC.LT                COMCIDP    1412    A
COMCIDP   BRK      KEYW   (BRGE BRAD,A1,A2,LL,UL,INC),BTC.GE                COMCIDP    1413    A
COMCIDP   BRK      KEYW   (BRLE BRAD,A1,A2,LL,UL,INC),BTC.LE                COMCIDP    1414    A
COMCIDP   BRK      KEYW   (BRGT BRAD,A1,A2,LL,UL,INC),BTC.GT                COMCIDP    1415    A
COMCIDP                                                                     COMCIDP    1416    A
COMCIDP  *         *STEP* KEYWORDS.                                         COMCIDP    1417    A
COMCIDP  *                                                                  COMCIDP    1418    A
COMCIDP  *         NOTE THAT THE *STEP* TYPE CODE (XTC) IS PLACED IN THE *XAT*  COMCIDP    1419    A
COMCIDP  *         FIELD OF EACH *STEP* KEYWORD.                            COMCIDP    1420    A
COMCIDP  *                                                                  COMCIDP    1421    A
COMCIDP  *         *STEP* TYPE CODES (XTC) HAVE THE FOLLOWING SIGNIFICANCE --  COMCIDP    1422    A
COMCIDP  *                                                                  COMCIDP    1423    A
COMCIDP  *                XTC = 0  IS A STEP WITH NO ADDR EXPRESSION--      COMCIDP    1424    A
COMCIDP  *                            STEP L,RJ                             COMCIDP    1425    A
COMCIDP  *                XTC .LT. XTC.EQ ARE STEPS WITH 1 ADDR EXPRESSION--  COMCIDP    1426    A
COMCIDP  *                            STMI ADDR,L,RJ                        COMCIDP    1427    A
COMCIDP  *                XTC .GE. XTC.EQ AND                               COMCIDP    1428    A
COMCIDP  *                    .LT. XTC.RNG ARE STEPS WITH 2 ADDR EXPRESSIONS--  COMCIDP    1429    A
COMCIDP  *                            STGT ADDR1,ADDR2,L,RJ                 COMCIDP    1430    A
COMCIDP  *                XTC .GE. XTC.RNG ARE STEPS THAT ARE SPECIAL CASED--  COMCIDP    1431    A
COMCIDP  *                            STAR ADDR,L,RJ                        COMCIDP    1432    A
COMCIDP  *                                                                  COMCIDP    1433    A
COMCIDP  *           (ALSO SEE *BREAK* KEYWORDS)                            COMCIDP    1434    A
COMCIDP                                                                     COMCIDP    1435    A
COMCIDP   STP      KEYW   (STEP L,RJ),0                                     COMCIDP    1436    A
COMCIDP   STP      KEYW   (STPL A1,L,RJ),XTC.PL                             COMCIDP    1437    A
COMCIDP   STP      KEYW   (STMI A1,L,RJ),XTC.MI                             COMCIDP    1438    A
COMCIDP   STP      KEYW   (STZR A1,L,RJ),XTC.ZR                             COMCIDP    1439    A
COMCIDP   STP      KEYW   (STNZ A1,L,RJ),XTC.NZ                             COMCIDP    1440    A
COMCIDP   STP      KEYW   (STEQ A1,A2,L,RJ),XTC.EQ                          COMCIDP    1441    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP   STP      KEYW   (STNE A1,A2,L,RJ),XTC.NE                                COMCIDP    1442    A
COMCIDP   STP      KEYW   (STLT A1,A2,L,RJ),XTC.LT                                COMCIDP    1443    A
COMCIDP   STP      KEYW   (STGE A1,A2,L,RJ),XTC.GE                                COMCIDP    1444    A
COMCIDP   STP      KEYW   (STLE A1,A2,L,RJ),XTC.LE                                COMCIDP    1445    A
COMCIDP   STP      KEYW   (STGT A1,A2,L,RJ),XTC.GT                                COMCIDP    1446    A
COMCIDP   STP      KEYW   (STRANGE A1,A2,L,RJ),XTC.RNG                            COMCIDP    1447    A
COMCIDP   XNR      KEYW   (STNR NR,L,RJ),XTC.NR                                   COMCIDP    1448    A
COMCIDP   STP      KEYW   (STAR A1,A2,L,RJ),XTC.AR                                COMCIDP    1449    A
COMCIDP            DATA   0           END OF TABLE MARK                           COMCIDP    1450    A
COMCIDP                                                                           COMCIDP    1451    A
COMCIDP                                                                           COMCIDP    1452    A
COMCIDP   FW=UKT   DATA   0           ** LINK WORD **                             COMCIDP    1453    A
COMCIDP   IDPKSTO  SPACE  4,10                                                    COMCIDP    1454    A
COMCIDP  **        *STEP* OPTIONS KEYWORD TABLE.                                  COMCIDP    1455    A
COMCIDP  *                                                                        COMCIDP    1456    A
COMCIDP  *           THERE IS A ONE-TO-ONE RELATIONSHIP BETWEEN THE STEP OPTIONS  COMCIDP    1457    A
COMCIDP  *           KEYWORD TABLE AND THE STEP OPTIONS FLAGS (SEE *IDPXOP*)      COMCIDP    1458    A
COMCIDP                                                                           COMCIDP    1459    A
COMCIDP   IDPKSTO  BSS    0                                                       COMCIDP    1460    A
COMCIDP            SUBKEY L,1                                                     COMCIDP    1461    A
COMCIDP            SUBKEY RJ,1                                                    COMCIDP    1462    A
COMCIDP            DATA   0           END OF TABLE MARK                           COMCIDP    1463    A
COMCIDP   KEY      SPACE  4,10                                                    COMCIDP    1464    A
COMCIDP  **        *OUTPUT* OPTIONS KEYWORD TABLE.                                COMCIDP    1465    A
COMCIDP                                                                           COMCIDP    1466    A
COMCIDP   IDPKOUT  BSS    0                                                       COMCIDP    1467    A
COMCIDP            SUBKEY B,ST=OUT3                                               COMCIDP    1468    A
COMCIDP            SUBKEY BECHO,ST=OUT4                                           COMCIDP    1469    A
COMCIDP            SUBKEY I,ST=OUT2                                               COMCIDP    1470    A
COMCIDP            SUBKEY IECHO,ST=OUT5                                           COMCIDP    1471    A
COMCIDP            DATA   0           END OF TABLE MARK                           COMCIDP    1472    A
COMCIDP   FW.KOPT  SPACE  4,10                                                    COMCIDP    1473    A
COMCIDP  **        FW.KOPT - *OPTION* SUBKEYWORD TABLE.                           COMCIDP    1474    A
COMCIDP  *                                                                        COMCIDP    1475    A
COMCIDP  *                                                                        COMCIDP    1476    A
COMCIDP                                                                           COMCIDP    1477    A
COMCIDP                                                                           COMCIDP    1478    A
COMCIDP   FW.KOPT  BSS    0           ** FWA **                                   COMCIDP    1479    A
COMCIDP            SUBKEY MEM,ST=OPT2                                             COMCIDP    1480    A
COMCIDP            SUBKEY XJR,ST=OPT3                                             COMCIDP    1481    A
COMCIDP            DATA   0           ** EOT **                                   COMCIDP    1482    A
COMCIDP   ERROR    SPACE  10,20                                                   COMCIDP    1483    A
COMCIDP  ***       IDP ERROR MESSAGES.                                            COMCIDP    1484    A
COMCIDP  *                                                                        COMCIDP    1485    A
COMCIDP  *                                                                        COMCIDP    1486    A
COMCIDP                                                                           COMCIDP    1487    A
COMCIDP                                                                           COMCIDP    1488    A
COMCIDP   FW.ERR   BSS    0           ** FWA **                                   COMCIDP    1489    A
COMCIDP   E.ADDR   ERROR  (ADDR OUT-OF-RANGE)                                     COMCIDP    1490    A
COMCIDP   E.SC     ERROR  (STMT KEYWORD PRECEDED BY UNKNOWN CHARACTER)            COMCIDP    1491    A
COMCIDP   E.KEY    ERROR  (UNKNOWN KEYWORD)                                       COMCIDP    1492    A
COMCIDP   E.BRAD   ERROR  (*BREAK* ADDR IS OUT-OF-RANGE)                          COMCIDP    1493    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP   E.BRA1   ERROR  (*ADDR1* IS OUT-OF-RANGE)                          COMCIDP   1494   A
COMCIDP   E.BRA2   ERROR  (*ADDR2* IS OUT-OF-RANGE)                          COMCIDP   1495   A
COMCIDP   E.FLL    ERROR  (FWA, LWA, OR LEN OUT-OF-RANGE)                    COMCIDP   1496   A
COMCIDP   E.LFN    ERROR  (YOU MUST SPECIFY AN LFN)                          COMCIDP   1497   A
COMCIDP   E.JP     ERROR  (*JUMP* ADDR OUT-OF-RANGE)                         COMCIDP   1498   A
COMCIDP   E.REG    ERROR  (YOU SPECIFIED A NON-REGISTER NAME)                COMCIDP   1499   A
COMCIDP   E.SET    ERROR  (YOU MUST SPECIFY A *SET* NAME)                    COMCIDP   1500   A
COMCIDP   E.STO    ERROR  (*STORE* ADDR IS MISSING OR OUT-OF-RANGE)          COMCIDP   1501   A
COMCIDP   E.POP    ERROR  (PARSER ENCOUNTERED ILLEGAL TOKEN/OPERATOR/OPERAND) COMCIDP   1502   A
COMCIDP   E.VAR    ERROR  (YOU SPECIFIED AN UNKNOWN/UNDEFINED SYMBOL NAME)   COMCIDP   1503   A
COMCIDP   E.CON    ERROR  (ERROR DURING CONSTANT CONVERSION)                 COMCIDP   1504   A
COMCIDP   E.IND    ERROR  (INDIRECT ADDR OUT-OF-RANGE)                       COMCIDP   1505   A
COMCIDP   E.OPT    ERROR  (YOU JUST SPECIFIED AN UNKNOWN OPTION)             COMCIDP   1506   A
COMCIDP   E.PAST   ERROR  (PARSER COULD NOT BALANCE OPERATORS AND OPERANDS)  COMCIDP   1507   A
COMCIDP   E.STB    ERROR  (STMT TOO BIG FOR PARSER.  PLEASE SIMPLIFY)        COMCIDP   1508   A
COMCIDP   E.SYTL   ERROR  (SYMBOL NAME TOO LONG)                             COMCIDP   1509   A
COMCIDP   E.UNOP   ERROR  (YOU JUST TRIED TO USE A BINARY OPERATOR AS A UNARY) COMCIDP   1510   A
COMCIDP                                                                      COMCIDP   1511   A
COMCIDP   L.ERR    =      *-FW.ERR                                           COMCIDP   1512   A
COMCIDP   SYSER    SPACE  4,10                                               COMCIDP   1513   A
COMCIDP  ***       IDP *SYSTEM* ERRORS.                                      COMCIDP   1514   A
COMCIDP  *                                                                   COMCIDP   1515   A
COMCIDP  *                                                                   COMCIDP   1516   A
COMCIDP  *         THE FOLLOWING ARE THE UNSUPPRESSABLE *SYSTEM* TYPE        COMCIDP   1517   A
COMCIDP  *         ERRORS THAT IDP OUTPUTS WHEN IT DETECTS AN INTERNAL       COMCIDP   1518   A
COMCIDP  *         ERROR.                                                    COMCIDP   1519   A
COMCIDP                                                                      COMCIDP   1520   A
COMCIDP                                                                      COMCIDP   1521   A
COMCIDP   FW.SER   BSS    0           ** FWA **                              COMCIDP   1522   A
COMCIDP   SE.KEY   SYSERR (ONE OF IDP'S KEYWORD TABLES IS OUT-OF-RANGE)      COMCIDP   1523   A
COMCIDP   SE.CON   SYSERR (PPU PGM *CON* DETECTED AN ERROR W/ CONNECT OR DISCONT) COMCIDP   1524   A
COMCIDP   SE.FLL   SYSERR (SOMEONE SCREWED UP YOUR FWA, LWA, LEN.  WILD STORE?) COMCIDP   1525   A
COMCIDP   SE.TOV   SYSERR (TABLE OVERFLOW.  GIVE UP AN ENTRY, TRY AGAIN)     COMCIDP   1526   A
COMCIDP   SE.PAD   SYSERR (ADDR OF PROCESSOR FOR THIS KEYWORD IS OUT-OF-RANGE) COMCIDP   1527   A
COMCIDP   SE.PTR   SYSERR (NO POINTER DEFINED FOR THIS KEYWORD)              COMCIDP   1528   A
COMCIDP   SE.BPTR  SYSERR (BAD POINTER REF.  CHECK ENTRY REGS TO *PTR*)      COMCIDP   1529   A
COMCIDP   SE.TOK   SYSERR (ERROR DURING TOKEN GENERATION.  ENTIRE LINE TRASHED) COMCIDP   1530   A
COMCIDP   SE.UFT   SYSERR (*UFT=* ENTRY IS OUT-OF-RANGE)                     COMCIDP   1531   A
COMCIDP   SE.UNPL  SYSERR (*FRA/FAA* ERROR WITH UNARY PLUS)                  COMCIDP   1532   A
COMCIDP   SE.BRK   SYSERR (NO BREAK TABLE ENTRY FOR THIS BREAK.  YOUR MOVE...) COMCIDP   1533   A
COMCIDP   SE.NPP   SYSERR (NO PIP PROCESSOR)                                 COMCIDP   1534   A
COMCIDP   SE.RPN   SYSERR (RPN TABLE OVERFLOW)                               COMCIDP   1535   A
COMCIDP   SE.CONS  SYSERR (ERROR DURING CONSTANT CONVERSION)                 COMCIDP   1536   A
COMCIDP   L.SER    =      *-FW.SER                                           COMCIDP   1537   A
COMCIDP   IDPBA    SPACE  4,10                                               COMCIDP   1538   A
COMCIDP  **        IDPBA/IDPBC - BREAK ADDRESS AND BREAK CONTENTS PARALLEL TBLS. COMCIDP   1539   A
COMCIDP                                                                      COMCIDP   1540   A
COMCIDP   L.BA     =      10          LENGTH OF *IDPBA*                     COMCIDP   1541   A
COMCIDP   IDPBA    BSS    0           FWA OF *BREAK* ADDR TABLE             COMCIDP   1542   A
COMCIDP   .BA      DUP    L.BA,1                                            COMCIDP   1543   A
COMCIDP            CON    -1                                                COMCIDP   1544   A
COMCIDP            DATA   0           END OF TABLE TERMINATOR               COMCIDP   1545   A
```

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP   IDPBC    BSSZ   2*L.BA      *BREAK* ADDR CONTENTS TABLE                COMCIDP    1546    A
COMCIDP   IDPTMP   SPACE  4,10                                                   COMCIDP    1547    A
COMCIDP  **        IDP USER *TEMP* TABLE.                                        COMCIDP    1548    A
COMCIDP                                                                          COMCIDP    1549    A
COMCIDP   L.TMP    =      10D                                                    COMCIDP    1550    A
COMCIDP   IDPTMP   BSSZ   L.TMP                                                  COMCIDP    1551    A
COMCIDP   IDPSET   SPACE  4,10                                                   COMCIDP    1552    A
COMCIDP  **        IDP *SET* TABLE DEFINITION.                                   COMCIDP    1553    A
COMCIDP                                                                          COMCIDP    1554    A
COMCIDP   L.SET    =      30B         LENGTH OF IDP *SET* TABLE                  COMCIDP    1555    A
COMCIDP   IDPSET   BSS    0                                                      COMCIDP    1556    A
COMCIDP            DUP    L.SET,1                                                COMCIDP    1557    A
COMCIDP            CON    -1                                                     COMCIDP    1558    A
COMCIDP            DATA   0                                                      COMCIDP    1559    A
COMCIDP   IDPXFT   SPACE  4,10                                                   COMCIDP    1560    A
COMCIDP  **        IDPXFT - MOST RECENT TRANSFER ADDRESSES TABLE.                COMCIDP    1561    A
COMCIDP  *                                                                       COMCIDP    1562    A
COMCIDP  *           TABLE FORMAT --                                             COMCIDP    1563    A
COMCIDP  *                                                                       COMCIDP    1564    A
COMCIDP  *             XFT+0    VFD  30/-1  AVAILABLE ENTRY(ONLY IF XFT NOT FULL) COMCIDP    1565    A
COMCIDP  *                        .                                              COMCIDP    1566    A
COMCIDP  *                        .                                              COMCIDP    1567    A
COMCIDP  *               +N    VFD  30/POS COUNTER,30/P REG OF EARLIEST XFER     COMCIDP    1568    A
COMCIDP  *               +N+1  VFD  30/POS COUNTER,30/P REG OF LATER XFER       COMCIDP    1569    A
COMCIDP  *                        .                                              COMCIDP    1570    A
COMCIDP  *              +L.XFT VFD  60/0   END OF TABLE MARK                     COMCIDP    1571    A
COMCIDP                                                                          COMCIDP    1572    A
COMCIDP   L.XFT    =      4                                                      COMCIDP    1573    A
COMCIDP   IDPXFT   BSS    0                                                      COMCIDP    1574    A
COMCIDP            DUP    L.XFT,1                                                COMCIDP    1575    A
COMCIDP            CON    -1                                                     COMCIDP    1576    A
COMCIDP            DATA   0                                                      COMCIDP    1577    A
COMCIDP   IFR      SPACE  4,10                                                   COMCIDP    1578    A
COMCIDP  ***       IFR - IDP FREEZE RESTART.                                     COMCIDP    1579    A
COMCIDP  *                                                                       COMCIDP    1580    A
COMCIDP  *                                                                       COMCIDP    1581    A
COMCIDP                                                                          COMCIDP    1582    A
COMCIDP   FW.IFR   BSS    0           MARK FWA OF *FREEZE* RESTART AREA          COMCIDP    1583    A
COMCIDP                                                                          COMCIDP    1584    A
COMCIDP            QUAL   IFR                                                    COMCIDP    1585    A
COMCIDP   IFR      SPACE  4,10                                                   COMCIDP    1586    A
COMCIDP  **        IFR - IDP FREEZE RECOVERY/RESTART.                            COMCIDP    1587    A
COMCIDP  *                                                                       COMCIDP    1588    A
COMCIDP  *                                                                       COMCIDP    1589    A
COMCIDP  *         THIS ROUTINE IS USED TO RESTART (I.E. SWAPIN) AN              COMCIDP    1590    A
COMCIDP  *         IDP HOST PROGRAM THAT WAS FROZEN VIA *FREEZE*                 COMCIDP    1591    A
COMCIDP  *         COMMAND.  THE PROGRAM TO RESTART WAS WRITTEN BY *FRZ*         COMCIDP    1592    A
COMCIDP  *         TO LFN *F.FRZ* IN A SPECIAL FORMAT --                         COMCIDP    1593    A
COMCIDP  *                                                                       COMCIDP    1594    A
COMCIDP  *           RECORD 1  CONTAINS *IFR* (THE CODE YOU ARE LOOKING AT)      COMCIDP    1595    A
COMCIDP  *                     IN THE FORMAT OF AN ABSOLUTE BINARY.              COMCIDP    1596    A
COMCIDP  *                                                                       COMCIDP    1597    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *          RECORD 2  CONTAINS THE CORE IMAGE OF THE FROZEN PROGRAM.          COMCIDP    1598    A
COMCIDP  *                                                                            COMCIDP    1599    A
COMCIDP  *        BECAUSE RECORD 1 OF *F.FRZ* LOOKS LIKE AN ABS BINARY, THE           COMCIDP    1600    A
COMCIDP  *        PROGRAMMER NEED ONLY ENTER THE NAME OF THE FREEZE FILE AT           COMCIDP    1601    A
COMCIDP  *        HIS OR HER TERMINAL IN ORDER TO RESTART THE HOST.                   COMCIDP    1602    A
COMCIDP  *                                                                            COMCIDP    1603    A
COMCIDP  *        AFTER *IFR* HAS BEEN LOADED, IT WILL MAKE AN OPERATING SYSTEM       COMCIDP    1604    A
COMCIDP  *        REQUEST FOR THE AMOUNT OF CM THAT THE HOST HAD AT THE TIME IT       COMCIDP    1605    A
COMCIDP  *        WAS FROZEN PLUS ENOUGH FOR *IFR* TO PERFORM THE FOLLOWING           COMCIDP    1606    A
COMCIDP  *        TASKS.  AFTER *IFR* HAS CREATED A HOLE FOR THE HOST                 COMCIDP    1607    A
COMCIDP  *        BY *MEM*ING, IT WILL *PLUG* SOME CODE UP ABOVE THE HOLE,            COMCIDP    1608    A
COMCIDP  *        WHICH WILL READ IN THE 2ND RECORD OF *F.FRZ*.                       COMCIDP    1609    A
COMCIDP  *                                                                            COMCIDP    1610    A
COMCIDP  *        WE ALL HAVE CHET RICHARDS OF SVLOPS TO THANK FOR THIS               COMCIDP    1611    A
COMCIDP  *        DIABOLICALLY CLEVER IDEA.  HI HO SILVER, AND AWAY...                COMCIDP    1612    A
COMCIDP  *                                                                            COMCIDP    1613    A
COMCIDP  *        IMPORTANT NOTE--  IT IS IMPORTANT TO REALIZE THAT                   COMCIDP    1614    A
COMCIDP  *        THIS CODE, AS IT EXISTS IN /COMCIDP/, IS A DATA SECTION,            COMCIDP    1615    A
COMCIDP  *        I.E. IT CANNOT BE EXECUTED.  IT IS HERE ONLY SO THAT *FRZ*          COMCIDP    1616    A
COMCIDP  *        CAN WRITE IT OUT AS THE 1ST RECORD ON *F.FRZ*.                      COMCIDP    1617    A
COMCIDP  *                                                                            COMCIDP    1618    A
COMCIDP  *        ENTRY  NONE                                                         COMCIDP    1619    A
COMCIDP  *                                                                            COMCIDP    1620    A
COMCIDP  *        EXIT   TO *RHH* IN HIGH CORE TO READ IN FROZEN HOST                 COMCIDP    1621    A
COMCIDP  *                                                                            COMCIDP    1622    A
COMCIDP  *        USES   IRRELEVANT                                                   COMCIDP    1623    A
COMCIDP  *                                                                            COMCIDP    1624    A
COMCIDP  *        CALLS  MEMORY                                                       COMCIDP    1625    A
COMCIDP                                                                               COMCIDP    1626    A
COMCIDP                                                                               COMCIDP    1627    A
COMCIDP           LOC    RA.ORG                                                       COMCIDP    1628    A
COMCIDP                                                                               COMCIDP    1629    A
COMCIDP           VFD    12/5000B,12/0,18/RA.ORG,18/IFR                               COMCIDP    1630    A
COMCIDP                                                                               COMCIDP    1631    A
COMCIDP   IFR     BSS    0          ** LOADER ENTRY POINT **                          COMCIDP    1632    A
COMCIDP           SB1    1                                                            COMCIDP    1633    A
COMCIDP                                                                               COMCIDP    1634    A
COMCIDP  *        REQUEST ENOUGH MEMORY FOR THE FROZEN HOST PLUS *RHH*.               COMCIDP    1635    A
COMCIDP  *        ALSO, IF *CMM* WAS ACTIVE IN FROZEN HOST, INDICATE SO TO            COMCIDP    1636    A
COMCIDP  *        OPERATING SYSTEM.                                                   COMCIDP    1637    A
COMCIDP                                                                               COMCIDP    1638    A
COMCIDP           SA2    MEM#HOST    (X2) = 30/AFLS OF FROZEN HOST,30/OTHER           COMCIDP    1639    A
COMCIDP           SX6    L.RHH+L.IFRSZ+10B                                            COMCIDP    1640    A
COMCIDP           LX6    30                                                           COMCIDP    1641    A
COMCIDP           IX6    X2+X6       (X6) = 30/ENOUGH CORE FOR FROZEN HOST            COMCIDP    1642    A
COMCIDP  *                                  AND *RHH*, 30/OTHER                       COMCIDP    1643    A
COMCIDP           SA6    MEM#RHH                                                      COMCIDP    1644    A
COMCIDP                                                                               COMCIDP    1645    A
COMCIDP                                                                               COMCIDP    1646    A
COMCIDP           SA1    IFRA        (X1) = *MEM* REQUEST WORD                        COMCIDP    1647    A
COMCIDP           BX6    X1                                                           COMCIDP    1648    A
COMCIDP           RJ     SYS         MAKE SYSTEM REQUEST                              COMCIDP    1649    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP                                                                                  COMCIDP     1650     A
COMCIDP  *          SET UP *F.FRZ* FET SO THAT A *READSKP* WILL READ                     COMCIDP     1651     A
COMCIDP  *          THE FROZEN HOST FROM *F.FRZ/RECORD 2* INTO                           COMCIDP     1652     A
COMCIDP  *          THE HOLE CREATED BY *MEM*.                                           COMCIDP     1653     A
COMCIDP                                                                                  COMCIDP     1654     A
COMCIDP             SA1    RA.PGN                                                        COMCIDP     1655     A
COMCIDP             AX2    30                                                            COMCIDP     1656     A
COMCIDP             SX7    X2          (X7) = OUT = AFLS OF FROZEN HOST                  COMCIDP     1657     A
COMCIDP             MX3    60-18                                                         COMCIDP     1658     A
COMCIDP             SA7    F.FRZ+3     OUT = AFLS OF FROZEN HOST                         COMCIDP     1659     A
COMCIDP             SX4    122B        (X4) = *OPEN/NO REWIND* CIO CODE                  COMCIDP     1660     A
COMCIDP             BX6    X3*X1                                                         COMCIDP     1661     A
COMCIDP             SX7    X7+B1       (X7) = LIMIT = AFLS+1 OF FROZEN HOST              COMCIDP     1662     A
COMCIDP             IX6    X6+X4                                                         COMCIDP     1663     A
COMCIDP             SA7    A7+B1       LIMIT = AFLS+1 OF FROZEN HOST                     COMCIDP     1664     A
COMCIDP             SA6    F.FRZ                                                         COMCIDP     1665     A
COMCIDP             =X7    RA.MTR+1                                                      COMCIDP     1666     A
COMCIDP             SA7    A6+B1       FIRST = RA.MTR+1                                  COMCIDP     1667     A
COMCIDP             SA7    A7+B1       IN = FIRST =RA.MTR+1                              COMCIDP     1668     A
COMCIDP                                                                                  COMCIDP     1669     A
COMCIDP  *          MOVE *RHH* AND *F.FRZ* FET TO HIGH CORE                              COMCIDP     1670     A
COMCIDP  *          (ABOVE HOLE CREATED FOR FROZEN HOST).                                COMCIDP     1671     A
COMCIDP                                                                                  COMCIDP     1672     A
COMCIDP             SB4    B0                                                            COMCIDP     1673     A
COMCIDP             SA1    FW.RHH      (A1,X1) = A+C OF 1ST WORD TO MOVE                 COMCIDP     1674     A
COMCIDP             SB3    L.RHH       (B2) = NR OF WORDS TO MOVE                        COMCIDP     1675     A
COMCIDP             SB2    X2+L.IFRSZ  (B2) = ADDR TO MOVE TO                            COMCIDP     1676     A
COMCIDP                                                                                  COMCIDP     1677     A
COMCIDP   IFR2      BX6    X1                                                            COMCIDP     1678     A
COMCIDP             SA6    B2+B4                                                         COMCIDP     1679     A
COMCIDP             SB4    B4+B1                                                         COMCIDP     1680     A
COMCIDP             SA1    A1+B1                                                         COMCIDP     1681     A
COMCIDP             LT     B4,B3,IFR2  IF NOT DONE                                       COMCIDP     1682     A
COMCIDP                                                                                  COMCIDP     1683     A
COMCIDP  *          OPEN *F.FRZ*.                                                        COMCIDP     1684     A
COMCIDP                                                                                  COMCIDP     1685     A
COMCIDP             SA1    IFRB        (X1) = *CIO* REQUEST WORD                         COMCIDP     1686     A
COMCIDP             SX2    B2+F.FRZ-FW.RHH   (X2) = FWA OF *F.FRZ* FET                  COMCIDP     1687     A
COMCIDP             BX6    X1+X2                                                         COMCIDP     1688     A
COMCIDP             SA6    A1          SAVE *CIO* REQUEST W/ ADDR OF MOVED *F.FRZ*       COMCIDP     1689     A
COMCIDP             RJ     SYS         MAKE SYSTEM REQUEST TO OPEN *F.FRZ*               COMCIDP     1690     A
COMCIDP                                                                                  COMCIDP     1691     A
COMCIDP  *          SET UP SYSTEM REQUEST TO PERFORM THE *READSKP*                       COMCIDP     1692     A
COMCIDP  *          WHICH WILL READ THE FROZEN HOST INTO THE HOLE CREATED                COMCIDP     1693     A
COMCIDP  *          BY *MEM*.                                                            COMCIDP     1694     A
COMCIDP                                                                                  COMCIDP     1695     A
COMCIDP   IFR3      SA1    RA.MTR                                                        COMCIDP     1696     A
COMCIDP             NZ     X1,IFR3     IF (RA.MTR) NOT CLEAR, WAIT...                    COMCIDP     1697     A
COMCIDP                                                                                  COMCIDP     1698     A
COMCIDP             SA1    SYS1                                                          COMCIDP     1699     A
COMCIDP             SA2    B2+F.FRZ-FW.RHH   (A2,X2) = A+C OF *F.FRZ* FET WORD 1         COMCIDP     1700     A
COMCIDP             SA3    IFRB        (X3) = *CIO* REQUEST WORD                         COMCIDP     1701     A

            0         1         2         3         4         5         6         7         8
            12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            MX4    60-18                                                    COMCIDP    1702    A
COMCIDP            BX7    X1                                                       COMCIDP    1703    A
COMCIDP            LX1    59-56       (X1) = .MI. IF *EQ* INSTRUCTION              COMCIDP    1704    A
COMCIDP  *                                  = .PL. IF *XJ* INSTRUCTION              COMCIDP    1705    A
COMCIDP            BX6    X4*X2       CLEAR BITS 0 THRU 17 OF *F.FRZ* WORD 1       COMCIDP    1706    A
COMCIDP            SX4    22B         (X4) = *READSKP* CIO CODE                    COMCIDP    1707    A
COMCIDP            IX6    X6+X4       MERGE *READSKP* CIO CODE                     COMCIDP    1708    A
COMCIDP            SA6    A2                                                       COMCIDP    1709    A
COMCIDP            BX6    X3          (X6) = *CIO* REQUEST WORD                    COMCIDP    1710    A
COMCIDP                                                                            COMCIDP    1711    A
COMCIDP  *         RELOCATE *WAIT (RA.MTR) LOOP* IF NOT USING                      COMCIDP    1712    A
COMCIDP  *         *XJ* INSTRUCTION.                                               COMCIDP    1713    A
COMCIDP                                                                            COMCIDP    1714    A
COMCIDP            PL     X1,IFR4     IF USING *XJ*                                COMCIDP    1715    A
COMCIDP            MX2    -18                                                      COMCIDP    1716    A
COMCIDP            SX3    B2+B1       (X3) = ADDR OF *RHH* INSTRUCTIONS WHICH      COMCIDP    1717    A
COMCIDP  *                                   ACTUALLY MAKES THE *CIO* REQUEST      COMCIDP    1718    A
COMCIDP  *                                   TO READ *F.FRZ*                       COMCIDP    1719    A
COMCIDP            BX7    X2*X7       CLEAR OLD ADDR                               COMCIDP    1720    A
COMCIDP            IX7    X7+X3       MERGE NEW ADDR                               COMCIDP    1721    A
COMCIDP            SA7    B2+B1                                                    COMCIDP    1722    A
COMCIDP                                                                            COMCIDP    1723    A
COMCIDP  *         EVERYTHING IS READY TO GO...                                    COMCIDP    1724    A
COMCIDP  *         TRANSFER CONTROL TO *RHH* UP IN HIGH CORE.                      COMCIDP    1725    A
COMCIDP                                                                            COMCIDP    1726    A
COMCIDP   IFR4     JP     B2+                                                      COMCIDP    1727    A
COMCIDP                                                                            COMCIDP    1728    A
COMCIDP                                                                            COMCIDP    1729    A
COMCIDP  *         *MEM* REQUEST WORD.                                             COMCIDP    1730    A
COMCIDP                                                                            COMCIDP    1731    A
COMCIDP   IFRA     VFD    18/3LMEM,3/2,21/0,18/MEM#RHH                             COMCIDP    1732    A
COMCIDP                                                                            COMCIDP    1733    A
COMCIDP  *         *CIO* REQUEST WORD.                                             COMCIDP    1734    A
COMCIDP                                                                            COMCIDP    1735    A
COMCIDP   IFRB     VFD    18/3LCIO,3/2,21/0,18/0                                   COMCIDP    1736    A
COMCIDP   SYS      SPACE  4                                                        COMCIDP    1737    A
COMCIDP  ***       SYS - PROCESS SYSTEM REQUEST.                                   COMCIDP    1738    A
COMCIDP  *                                                                         COMCIDP    1739    A
COMCIDP  *         ENTRY  (X6) = SYSTEM REQUEST.                                   COMCIDP    1740    A
COMCIDP  *                                                                         COMCIDP    1741    A
COMCIDP  *         EXIT   REQUEST PROCESSED.                                       COMCIDP    1742    A
COMCIDP  *                                                                         COMCIDP    1743    A
COMCIDP  *         USES   X - 1.                                                   COMCIDP    1744    A
COMCIDP  *                B - NONE.                                                COMCIDP    1745    A
COMCIDP  *                A - 1, 6.                                                COMCIDP    1746    A
COMCIDP  *                                                                         COMCIDP    1747    A
COMCIDP  *         CALLS  NONE.                                                    COMCIDP    1748    A
COMCIDP                                                                            COMCIDP    1749    A
COMCIDP                                                                            COMCIDP    1750    A
COMCIDP  *         INSTRUCTION WORDS FOR MONITOR CALL.                             COMCIDP    1751    A
COMCIDP                                                                            COMCIDP    1752    A
COMCIDP   SYSA     BSS    0                                                        COMCIDP    1753    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            LOC    *+2                                                    COMCIDP    1754    A
COMCIDP   +        SA1    A1          WAIT (RA+1) CLEAR IF AUTO RECALL           COMCIDP    1755    A
COMCIDP            LX1    59-40                                                  COMCIDP    1756    A
COMCIDP            NG     X1,*                                                   COMCIDP    1757    A
COMCIDP            LOC    SYSA+1                                                 COMCIDP    1758    A
COMCIDP                                                                           COMCIDP    1759    A
COMCIDP            XJ                                                             COMCIDP    1760    A
COMCIDP                                                                           COMCIDP    1761    A
COMCIDP   SYS1     EQ     SYS2        FIRST ENTRY                                COMCIDP    1762    A
COMCIDP                                                                           COMCIDP    1763    A
COMCIDP   SYS      PS                 ENTRY/EXIT                                 COMCIDP    1764    A
COMCIDP   +        SA1    1           WAIT (RA+1) CLEAR                          COMCIDP    1765    A
COMCIDP            NZ     X1,*                                                   COMCIDP    1766    A
COMCIDP            SA6    A1          ENTER REQUEST                              COMCIDP    1767    A
COMCIDP            EQ     SYS1                                                   COMCIDP    1768    A
COMCIDP                                                                           COMCIDP    1769    A
COMCIDP   *        INITIAL ENTRY TO SET TYPE OF CALL.                            COMCIDP    1770    A
COMCIDP                                                                           COMCIDP    1771    A
COMCIDP   SYS2     SA1    SYSA        NO MEJ/CEJ OPTION                          COMCIDP    1772    A
COMCIDP            BX6    X1                                                     COMCIDP    1773    A
COMCIDP                                                                           COMCIDP    1774    A
COMCIDP            SA1    RA.CEJ                                                 COMCIDP    1775    A
COMCIDP            PL     X1,SYS3     IF NO MEJ/CEJ PRESENT                      COMCIDP    1776    A
COMCIDP            SA1    SYSA+1                                                 COMCIDP    1777    A
COMCIDP            BX6    X1                                                     COMCIDP    1778    A
COMCIDP   SYS3     BSS    0                                                      COMCIDP    1779    A
COMCIDP                                                                           COMCIDP    1780    A
COMCIDP            SA6    SYS1        SET MONITOR CALL                           COMCIDP    1781    A
COMCIDP            SA1    1           RESET (A1)                                 COMCIDP    1782    A
COMCIDP            BX6    X1                                                     COMCIDP    1783    A
COMCIDP            RJ     SYS1-1      CLEAR STACK                                COMCIDP    1784    A
COMCIDP   IFR      SPACE  4,10                                                   COMCIDP    1785    A
COMCIDP   MEM#RHH  BSSZ   1           30/AFLS OF HOST PLUS *RHH*,30/OTHER        COMCIDP    1786    A
COMCIDP                                                                           COMCIDP    1787    A
COMCIDP   MEM#HOST BSSZ   1           30/AFLS OF HOST,30/OTHER                   COMCIDP    1788    A
COMCIDP                                                                           COMCIDP    1789    A
COMCIDP   L.IFRSZ  =      10B         LEN OF SAFETY ZONE BETWEEN HOST AND *RHH*  COMCIDP    1790    A
COMCIDP   RHH      SPACE  4,10                                                   COMCIDP    1791    A
COMCIDP   **       RHH - READ HOST INTO HOLE.                                    COMCIDP    1792    A
COMCIDP   *                                                                      COMCIDP    1793    A
COMCIDP   *                                                                      COMCIDP    1794    A
COMCIDP   *        THIS ROUTINE MAKES THE ACTUAL OPERATING SYSTEM REQUEST        COMCIDP    1795    A
COMCIDP   *        TO READ THE FROZEN HOST FROM *F.FRZ/RECORD 2*                 COMCIDP    1796    A
COMCIDP   *        INTO THE HOLE CREATED BY *IFR*.                               COMCIDP    1797    A
COMCIDP   *                                                                      COMCIDP    1798    A
COMCIDP   *        ENTRY  (X6) = *CIO* REQUEST WORD                              COMCIDP    1799    A
COMCIDP   *                                                                      COMCIDP    1800    A
COMCIDP   *        EXIT   TO *IDP=IFR* IN UNFROZEN HOST                          COMCIDP    1801    A
COMCIDP   *                                                                      COMCIDP    1802    A
COMCIDP   *        USES                                                          COMCIDP    1803    A
COMCIDP   *                                                                      COMCIDP    1804    A
COMCIDP   *        CALLS  NONE                                                   COMCIDP    1805    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP                                                                  COMCIDP    1806    A
COMCIDP   FW.RHH   BSS    0           MARK FWA OF*RHH*                   COMCIDP    1807    A
COMCIDP                                                                  COMCIDP    1808    A
COMCIDP   RHH      BSS    0           ** ENTRY **                        COMCIDP    1809    A
COMCIDP            SA6    RA.MTR                                         COMCIDP    1810    A
COMCIDP                                                                  COMCIDP    1811    A
COMCIDP   *        THE FOLLOWING LOCATION WILL BE PLUGGED WITH            COMCIDP    1812    A
COMCIDP   *        A *WAIT (RA.MTR)* LOOP BY *IFR* IF NO *XJ* INSTRUCTION COMCIDP    1813    A
COMCIDP   *        AVAILABLE.                                             COMCIDP    1814    A
COMCIDP                                                                  COMCIDP    1815    A
COMCIDP   +        XJ                                                     COMCIDP    1816    A
COMCIDP                                                                  COMCIDP    1817    A
COMCIDP   *        THE INTERACTIVE HOST IS NOW BACK IN CORE... RE-ENTER *IDP*. COMCIDP 1818  A
COMCIDP                                                                  COMCIDP    1819    A
COMCIDP   +        EQ     /"IDP"/IDP=IFR                                 COMCIDP    1820    A
COMCIDP   F.FRZ    SPACE  4,10                                           COMCIDP    1821    A
COMCIDP   **       F.FRZ - FET FOR *FREEZE* FILE.                        COMCIDP    1822    A
COMCIDP                                                                  COMCIDP    1823    A
COMCIDP   F.FRZ    BSS    0           ** FWA OF FET **                   COMCIDP    1824    A
COMCIDP   L.FRZ    =      101B        LENGTH OF *F.FRZ* BUFFER FOR RECORD 1 WRITE COMCIDP 1825  A
COMCIDP   *                           (I.E. SMALLEST POSSIBLE, 1 PRU)    COMCIDP    1826    A
COMCIDP   FRZ      FILEB  /"IDP"/FW.IFR,L.FRZ                            COMCIDP    1827    A
COMCIDP                                                                  COMCIDP    1828    A
COMCIDP                                                                  COMCIDP    1829    A
COMCIDP   L.RHH    =      *-FW.RHH+1                                     COMCIDP    1830    A
COMCIDP                                                                  COMCIDP    1831    A
COMCIDP            LOC    *O                                             COMCIDP    1832    A
COMCIDP            QUAL   *                                              COMCIDP    1833    A
COMCIDP                                                                  COMCIDP    1834    A
COMCIDP   L.IFR    =      *-FW.IFR+1  LENGTH OF IDP FREEZE RESTART PROGRAM COMCIDP 1835  A
COMCIDP   IFR      SPACE  4,10                                           COMCIDP    1836    A
COMCIDP   **       REDEFINE SYMBOLS THAT *IDP* NEEDS TO REFERENCE INSIDE /IFR/. COMCIDP 1837  A
COMCIDP                                                                  COMCIDP    1838    A
COMCIDP   F.FRZ    =      FW.IFR+/IFR/F.FRZ-RA.ORG                       COMCIDP    1839    A
COMCIDP   MEM#RHH  =      FW.IFR+/IFR/MEM#RHH-RA.ORG                     COMCIDP    1840    A
COMCIDP   MEM#HOST =      FW.IFR+/IFR/MEM#HOST-RA.ORG                    COMCIDP    1841    A
COMCIDP   IDP      TITLE  IDP "IDPVER"/THE EXECUTIVES.                   COMCIDP    1842    A
COMCIDP   IDP      SPACE  4,8                                            COMCIDP    1843    A
COMCIDP   **       IDP - INTERACTIVE DEBUGGING PACKAGE.                  COMCIDP    1844    A
COMCIDP   *                                                              COMCIDP    1845    A
COMCIDP   *                                                              COMCIDP    1846    A
COMCIDP   *               THIS IS THE INTERACTIVE DEBUGGING PACKAGE WHICH ALLOWS COMCIDP 1847  A
COMCIDP   *        ITS USER THE JOYS AND SORROWS OF INTERACTIVE DEBUGGING. COMCIDP    1848    A
COMCIDP   *                                                              COMCIDP    1849    A
COMCIDP   *               INITIAL ENTRY IS VIA *BREAK* MACRO (COMAIDP), WHICH COMCIDP 1850  A
COMCIDP   *        ASSEMBLES AN *RJ IDP=* AT THE DESIRED LOCATION TO BE BREAK- COMCIDP 1851  A
COMCIDP   *        POINTED. SUBSEQUENT ENTRIES CAN BE VIA MORE ASSEMBLED *BREAK*S COMCIDP 1852  A
COMCIDP   *        OR BY PLACING AN *IDP* GENERATED BREAKPOINT, VIA *BREAK* COMCIDP   1853    A
COMCIDP   *        COMMAND, AT THE DESIRED LOCATION TO BE BREAKPOINTED.  COMCIDP      1854    A
COMCIDP   *                                                              COMCIDP    1855    A
COMCIDP   *        FOR A LIST OF IDP'S OWN (I.E. NOT USER KEYWORDS), SEE IDP'S COMCIDP 1856  A
COMCIDP   *        KEYWORD LIST AT *FW.KEY* ET SEQ.                      COMCIDP    1857    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *                                                                  COMCIDP    1858    A
COMCIDP  *         MAY THE GODS BE WITH YOU...                              COMCIDP    1859    A
COMCIDP  *                                                                  COMCIDP    1860    A
COMCIDP  *         ENTRY  NONE                                              COMCIDP    1861    A
COMCIDP  *                                                                  COMCIDP    1862    A
COMCIDP  *         EXIT   NONE                                              COMCIDP    1863    A
COMCIDP  *                                                                  COMCIDP    1864    A
COMCIDP  *         USES   NONE        (OR IT WOULDNT BE ANY GOOD)           COMCIDP    1865    A
COMCIDP  *                                                                  COMCIDP    1866    A
COMCIDP  *         CALLS  (EXTERNAL TO COMCIDP)--                           COMCIDP    1867    A
COMCIDP  *                CDD,COD,DXB,OPEN,READC,RSR=,SFN,SVR=,WRITEC,WRITER COMCIDP    1868    A
COMCIDP                                                                     COMCIDP    1869    A
COMCIDP                                                                     COMCIDP    1870    A
COMCIDP                                                                     COMCIDP    1871    A
COMCIDP   IDP=     SUBR               ** ENTRY **                           COMCIDP    1872    A
COMCIDP   +        RJ     =XSVR       SAVE ALL REGISTERS                    COMCIDP    1873    A
COMCIDP   -        VFD    30/FW.SVR                                         COMCIDP    1874    A
COMCIDP                                                                     COMCIDP    1875    A
COMCIDP  *         SET UP *IDPFLG* - MASTER CONTROL FLAG.                   COMCIDP    1876    A
COMCIDP  *                                                                  COMCIDP    1877    A
COMCIDP  *           SET  XEC = *IDP=*                                      COMCIDP    1878    A
COMCIDP  *           IF 1ST TIME *IDP* HAS BEEN ENTERED (BIT *FTO* OFF), SET --  COMCIDP  1879  A
COMCIDP  *                FTO = 1ST TIME ONLY CODE HAS BEEN EXECUTED        COMCIDP    1880    A
COMCIDP  *                IDO = WRITING INTERACTIVELY (F.IDO)               COMCIDP    1881    A
COMCIDP  *                SNL = L.ISL                                       COMCIDP    1882    A
COMCIDP                                                                     COMCIDP    1883    A
COMCIDP            SA1    IDPFLG                                            COMCIDP    1884    A
COMCIDP            MX2    -IDF.XECL                                         COMCIDP    1885    A
COMCIDP            MX3    1                                                 COMCIDP    1886    A
COMCIDP            LX2    IDF.XECP-0                                        COMCIDP    1887    A
COMCIDP            BX6    X2*X1       CLEAR OLD *XEC*/SET NEW *XEC*         COMCIDP    1888    A
COMCIDP            MX4    -IDF.SNLL                                         COMCIDP    1889    A
COMCIDP            LX1    59-IDF.FTOP                                       COMCIDP    1890    A
COMCIDP            SA6    A1                                                COMCIDP    1891    A
COMCIDP            MI     X1,IDP2     IF NOT 1ST TIME                       COMCIDP    1892    A
COMCIDP                                                                     COMCIDP    1893    A
COMCIDP            LX4    IDF.SNLP-0                                        COMCIDP    1894    A
COMCIDP            LX3    IDF.FTOP-59                                       COMCIDP    1895    A
COMCIDP            BX6    X4*X6       CLEAR OLD *SNL*                       COMCIDP    1896    A
COMCIDP            =X1    L.ISL                                             COMCIDP    1897    A
COMCIDP            BX6    X6+X3       MERGE *FTO*                           COMCIDP    1898    A
COMCIDP            LX3    IDF.IDOP-IDF.FTOP                                 COMCIDP    1899    A
COMCIDP            BX6    X6+X1       MERGE NEW *SNL*                       COMCIDP    1900    A
COMCIDP            BX6    X6+X3       MERGE *IDO*                           COMCIDP    1901    A
COMCIDP            SA6    A1                                                COMCIDP    1902    A
COMCIDP                                                                     COMCIDP    1903    A
COMCIDP  *         SET UP LINK TO USER KEYWORD TABLE, *UKT=*.               COMCIDP    1904    A
COMCIDP                                                                     COMCIDP    1905    A
COMCIDP            SA1    FW=UKT      (X1) = LINK WORD TO USER KEYWORD TABLES COMCIDP    1906    A
COMCIDP            SX6    =YUKT=      (X6) = FWA OF 1ST USER KEYWORD TABLE  COMCIDP    1907    A
COMCIDP            NZ     X1,IDP1     IF USER HAS ALREADY LINKED TABLES THEMSELF COMCIDP 1908    A
COMCIDP            MI     X6,IDP1     IF USER DID NOT PROVIDE *UKT=*        COMCIDP    1909    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            SA6    A1+                                                  COMCIDP    1910    A
COMCIDP                                                                         COMCIDP    1911    A
COMCIDP  *         SET UP INTERACTIVE FILES.                                    COMCIDP    1912    A
COMCIDP                                                                         COMCIDP    1913    A
COMCIDP   IDP1     RJ     IIF         INITIALIZE INTERACTIVE FILES              COMCIDP    1914    A
COMCIDP                                                                         COMCIDP    1915    A
COMCIDP  *         CLEAR/INITIALIZE TABLES.                                     COMCIDP    1916    A
COMCIDP                                                                         COMCIDP    1917    A
COMCIDP            SB6    IDPBA                                                 COMCIDP    1918    A
COMCIDP            RJ     CLZ         CLEAR *IDP* TABLE -- BREAK ADDRESSES      COMCIDP    1919    A
COMCIDP            SB6    IDPSET                                                COMCIDP    1920    A
COMCIDP            RJ     CLZ         CLEAR *IDP* TABLE -- SET NAMES            COMCIDP    1921    A
COMCIDP            RJ     IST         (RE)INITIALIZE *SET* TABLE                COMCIDP    1922    A
COMCIDP            SB6    IDPXFT                                                COMCIDP    1923    A
COMCIDP            RJ     CLZ         CLEAR *IDP* TABLE -- XFER ADDRESSES       COMCIDP    1924    A
COMCIDP                                                                         COMCIDP    1925    A
COMCIDP  *         INITIAL BREAK PROCESSING.                                    COMCIDP    1926    A
COMCIDP                                                                         COMCIDP    1927    A
COMCIDP   IDP2     SA1    IDP=        (A1,X1) = A+C OF IDP ENTRY POINT          COMCIDP    1928    A
COMCIDP            RJ     BRK         BREAK PROCESSOR                           COMCIDP    1929    A
COMCIDP            ZR     X5,ST=END3  IF CONDITIONAL BREAK NOT SATISFIED        COMCIDP    1930    A
COMCIDP            RJ     FRK         CHECK FREQUENCY PARAMETERS                COMCIDP    1931    A
COMCIDP            ZR     X5,ST=END3  IF NO BREAK THIS TIME                     COMCIDP    1932    A
COMCIDP            RJ     UIO         USER IDP OWNCODE                          COMCIDP    1933    A
COMCIDP            PL     X1,ST=END3  IF NO INTERACTIVE BREAK THIS TIME         COMCIDP    1934    A
COMCIDP                                                                         COMCIDP    1935    A
COMCIDP  *         FLUSH *F.BDO - BATCH DEBUG OUTPUT FILE*.                     COMCIDP    1936    A
COMCIDP                                                                         COMCIDP    1937    A
COMCIDP            SB2    =YF.BDO                                               COMCIDP    1938    A
COMCIDP            RJ     FOF         FLUSH OUTPUT FILE                         COMCIDP    1939    A
COMCIDP                                                                         COMCIDP    1940    A
COMCIDP  *         INTRODUCE YOURSELF.                                          COMCIDP    1941    A
COMCIDP                                                                         COMCIDP    1942    A
COMCIDP   IDP2A    BSS    0                                                     COMCIDP    1943    A
COMCIDP            SA1    =10H **IDP**                                          COMCIDP    1944    A
COMCIDP            SA2    =10H CALLD BY                                         COMCIDP    1945    A
COMCIDP            SA3    X5+SN=HDR                                             COMCIDP    1946    A
COMCIDP            BX6    X1                                                    COMCIDP    1947    A
COMCIDP            LX7    X2                                                    COMCIDP    1948    A
COMCIDP            SA6    SNAPLNE                                               COMCIDP    1949    A
COMCIDP            SA7    A6+B1                                                 COMCIDP    1950    A
COMCIDP            BX6    X3                                                    COMCIDP    1951    A
COMCIDP            ZR     X3,IDP4     IF NO HEADER PROVIDED BY CALLER .OR.      COMCIDP    1952    A
COMCIDP  *                               IDP GENERATED BREAK                    COMCIDP    1953    A
COMCIDP            SA6    A6+B1                                                 COMCIDP    1954    A
COMCIDP            SA7    A6+B1                                                 COMCIDP    1955    A
COMCIDP                                                                         COMCIDP    1956    A
COMCIDP   IDP4     SA1    IDPPREG                                               COMCIDP    1957    A
COMCIDP            SX1    X1-1                                                  COMCIDP    1958    A
COMCIDP            RJ     FRA         FIND RELATIVE ADDRESS                     COMCIDP    1959    A
COMCIDP            PL     B7,IDP5     IF USER PROVIDED *FRA=*                   COMCIDP    1960    A
COMCIDP                                                                         COMCIDP    1961    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            RJ     =XCOD       CONVERT BINARY TO OCTAL DPC                    COMCIDP     1962     A
COMCIDP            BX6    X4                                                         COMCIDP     1963     A
COMCIDP            MX7    0                                                          COMCIDP     1964     A
COMCIDP                                                                              COMCIDP     1965     A
COMCIDP   IDP5     SA6    A7+B1                                                      COMCIDP     1966     A
COMCIDP            SA7    A6+B1                                                      COMCIDP     1967     A
COMCIDP            PRIDP  SNAPLNE                                                    COMCIDP     1968     A
COMCIDP   IDP=MN   SPACE  4,10                                                       COMCIDP     1969     A
COMCIDP   *        ** MAIN LOOP NODE **                                              COMCIDP     1970     A
COMCIDP                                                                              COMCIDP     1971     A
COMCIDP   IDP=MN   BSS    0                                                          COMCIDP     1972     A
COMCIDP            SA1    TB=STMT                                                    COMCIDP     1973     A
COMCIDP            PL     X1,IDP7     IF LAST STMT PROCESSED WAS NOT IN *PROC*       COMCIDP     1974     A
COMCIDP                                                                              COMCIDP     1975     A
COMCIDP   *        HERE IF LAST STATEMENT WAS IN A PROCEDURE.                        COMCIDP     1976     A
COMCIDP                                                                              COMCIDP     1977     A
COMCIDP            EQ     *+4S15      OOPS...                                        COMCIDP     1978     A
COMCIDP                                                                              COMCIDP     1979     A
COMCIDP   *        CHECK FOR AVAILABILITY OF NEXT STATEMENT.                         COMCIDP     1980     A
COMCIDP                                                                              COMCIDP     1981     A
COMCIDP   IDP7     SA2    X1          (X2) = LEN WORD FOR LAST STMT PROCESSED        COMCIDP     1982     A
COMCIDP            LX2    0-TB.LENP                                                  COMCIDP     1983     A
COMCIDP            SX1    X1                                                         COMCIDP     1984     A
COMCIDP            SX2    X2+B1                                                      COMCIDP     1985     A
COMCIDP            IX6    X1+X2       (X6) = ADDR OF LEN WORD FOR NEXT STMT          COMCIDP     1986     A
COMCIDP            SA6    TB=STMT                                                    COMCIDP     1987     A
COMCIDP            SA1    X6                                                         COMCIDP     1988     A
COMCIDP            SA5    X6+B1       (A5,X5) = A+C OF 1ST TOKEN IN NEXT STMT        COMCIDP     1989     A
COMCIDP            NZ     X1,IDP12    IF NOT END-OF-LINE IN TOKEN BUFFER             COMCIDP     1990     A
COMCIDP                                                                              COMCIDP     1991     A
COMCIDP   *        HERE TO READ NEW LINE.                                            COMCIDP     1992     A
COMCIDP                                                                              COMCIDP     1993     A
COMCIDP   IDP8     SA1    IDPFLG                                                     COMCIDP     1994     A
COMCIDP            BX2    X1                                                         COMCIDP     1995     A
COMCIDP            LX1    59-IDF.INPP                                                COMCIDP     1996     A
COMCIDP            LX2    59-IDF.BRFP                                                COMCIDP     1997     A
COMCIDP            BX6    X1+X2                                                      COMCIDP     1998     A
COMCIDP            MI     X6,IDP8A    IF READING FROM BATCH INPUT (F.BDI) .OR.       COMCIDP     1999     A
COMCIDP   *                              IN *BRIEF* MODE                             COMCIDP     2000     A
COMCIDP                                                                              COMCIDP     2001     A
COMCIDP            DUP    L.PRB,1                                                    COMCIDP     2002     A
COMCIDP            WRITEC =XF.IDO,(=C=  =)                                           COMCIDP     2003     A
COMCIDP                                                                              COMCIDP     2004     A
COMCIDP            WRITEC =XF.IDO,(=C= "PROMPT"=)                                    COMCIDP     2005     A
COMCIDP                                                                              COMCIDP     2006     A
COMCIDP   IDP8A    SB2    F.IDO                                                      COMCIDP     2007     A
COMCIDP            RJ     FOF         FLUSH OUTPUT FILE                              COMCIDP     2008     A
COMCIDP                                                                              COMCIDP     2009     A
COMCIDP   IDP9     RJ     RIL         READ IDP LINE                                  COMCIDP     2010     A
COMCIDP            SA2    IDPFLG                                                     COMCIDP     2011     A
COMCIDP            SA5    FW.LIM      (A5,X5) = A+C OF 1ST WORD IN SOURCE LINE       COMCIDP     2012     A
COMCIDP            ZR     X1,IDP10    IF EOR/EOF NOT ENCOUNTERED                     COMCIDP     2013     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            MX3    1                                                    COMCIDP    2014    A
COMCIDP            LX3    IDF.RAPP-59                                          COMCIDP    2015    A
COMCIDP            BX6    -X3*X2      CLEAR *RAP*                              COMCIDP    2016    A
COMCIDP            LX3    IDF.INPP-IDF.RAPP                                    COMCIDP    2017    A
COMCIDP            BX6    -X3*X6      CLEAR *INP*                              COMCIDP    2018    A
COMCIDP            LX2    59-IDF.INPP                                          COMCIDP    2019    A
COMCIDP            SA6    A2+                                                  COMCIDP    2020    A
COMCIDP            PL     X2,ST=STA   IF INTERACTIVE INPUT, EOR IS *STATUS*... COMCIDP    2021    A
COMCIDP            EQ     IDP8                                                 COMCIDP    2022    A
COMCIDP                                                                        COMCIDP    2023    A
COMCIDP  *         HERE TO ECHO LINE TO BATCH DEBUG OUTPUT FILE (F.BDO),       COMCIDP    2024    A
COMCIDP  *           IF REQUESTED.                                             COMCIDP    2025    A
COMCIDP                                                                        COMCIDP    2026    A
COMCIDP   IDP10    BX3    X2                                                   COMCIDP    2027    A
COMCIDP            LX2    59-IDF.BKOP                                          COMCIDP    2028    A
COMCIDP            LX3    59-IDF.BDOP                                          COMCIDP    2029    A
COMCIDP            BX4    X2*X3                                                COMCIDP    2030    A
COMCIDP            PL     X4,IDP10A   IF NO ECHO REQUESTED .OR. NOT WRITING TO COMCIDP    2031    A
COMCIDP  *                               BATCH DEBUG OUTPUT FILE (F.BDO)       COMCIDP    2032    A
COMCIDP            PRBDO  (=C=  =),1                                           COMCIDP    2033    A
COMCIDP            SA2    L=LIM                                                COMCIDP    2034    A
COMCIDP            PRBDO  FW.LIM-1,X2                                          COMCIDP    2035    A
COMCIDP                                                                        COMCIDP    2036    A
COMCIDP  *         HERE TO ECHO LINE TO INTERACTIVE DEBUG OUTPUT FILE (F.IDO), COMCIDP    2037    A
COMCIDP  *           IF REQUESTED.                                             COMCIDP    2038    A
COMCIDP                                                                        COMCIDP    2039    A
COMCIDP   IDP10A   SA1    IDPFLG                                               COMCIDP    2040    A
COMCIDP            BX2    X1                                                   COMCIDP    2041    A
COMCIDP            LX1    59-IDF.IKOP                                          COMCIDP    2042    A
COMCIDP            LX2    59-IDF.IDOP                                          COMCIDP    2043    A
COMCIDP            BX3    X1*X2                                                COMCIDP    2044    A
COMCIDP            PL     X3,IDP11    IF NO ECHO REQUESTED .OR. NOT WRITING TO COMCIDP    2045    A
COMCIDP  *                               INTERACTIVE DEBUG OUTPUT FILE (F.IDO) COMCIDP    2046    A
COMCIDP                                                                        COMCIDP    2047    A
COMCIDP            DUP    L.PRB,1                                              COMCIDP    2048    A
COMCIDP            WRITEC =XF.IDO,(=C=  =)                                     COMCIDP    2049    A
COMCIDP                                                                        COMCIDP    2050    A
COMCIDP            WRITEC =XF.IDO,FW.LIM-1                                     COMCIDP    2051    A
COMCIDP                                                                        COMCIDP    2052    A
COMCIDP  *         HERE WHEN READY TO ENTOKEN LINE.                            COMCIDP    2053    A
COMCIDP  *                                                                     COMCIDP    2054    A
COMCIDP  *           1. SET UP *TOKCOM*, THE *USER/TOK* COMMUNICATIONS AREA.   COMCIDP    2055    A
COMCIDP  *                                                                     COMCIDP    2056    A
COMCIDP  *           2. CALL *COMCTOK* (COMMON TOKEN GENERATOR) TO ENTOKEN     COMCIDP    2057    A
COMCIDP  *              THIS LINE.                                             COMCIDP    2058    A
COMCIDP                                                                        COMCIDP    2059    A
COMCIDP   IDP11    SX6    TOM=BOL     (X6) = ADDR OF 1ST *TOGEL* INST TO XEQ   COMCIDP    2060    A
COMCIDP            SX7    FW.LIM      (X7) = FWA OF LINE IMAGE TO ENTOKEN      COMCIDP    2061    A
COMCIDP            SA6    TC=P                                                 COMCIDP    2062    A
COMCIDP            SA7    TC=SOA                                               COMCIDP    2063    A
COMCIDP                                                                        COMCIDP    2064    A
COMCIDP            SA1    X7                                                   COMCIDP    2065    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            BX6    X1                                                  COMCIDP    2066    A
COMCIDP            SA6    TC=SOX                                              COMCIDP    2067    A
COMCIDP            SX6    -1                                                  COMCIDP    2068    A
COMCIDP            SX7    IDPTB       (X7) = FWA OF TOKEN BUFFER              COMCIDP    2069    A
COMCIDP            SA6    TC=SOC      SET TO *NO USEABLE CHAR IN (X4)*        COMCIDP    2070    A
COMCIDP            SA7    TC=TBA                                              COMCIDP    2071    A
COMCIDP            SA7    TB=STMT                                             COMCIDP    2072    A
COMCIDP                                                                       COMCIDP    2073    A
COMCIDP            SB2    FW.TCOM     (B2) = FWA OF *TOKCOM*                  COMCIDP    2074    A
COMCIDP            RJ     =XTOK       TOKEN GENERATOR                         COMCIDP    2075    A
COMCIDP            SX6    IDPTB                                               COMCIDP    2076    A
COMCIDP            SA5    IDPTB+1                                             COMCIDP    2077    A
COMCIDP            SA6    TB=STMT                                             COMCIDP    2078    A
COMCIDP            EQ     IDP12                                               COMCIDP    2079    A
COMCIDP                                                                       COMCIDP    2080    A
COMCIDP  *         BACK FROM *STATEMENT CONTROL* PROCESSING.  INCREMENT      COMCIDP    2081    A
COMCIDP  *         TOKEN BUFFER POINTER.                                     COMCIDP    2082    A
COMCIDP                                                                       COMCIDP    2083    A
COMCIDP   IDP=SC   BSS    0           ** ENTRY **                             COMCIDP    2084    A
COMCIDP            SA5    A5+1                                                COMCIDP    2085    A
COMCIDP                                                                       COMCIDP    2086    A
COMCIDP  *         HERE WITH A NEW STATEMENT TO BE PROCESSED.                COMCIDP    2087    A
COMCIDP  *                                                                    COMCIDP    2088    A
COMCIDP  *           1. CHECK FOR AND PROCESS ANY *STMT CONTROL* TOKENS      COMCIDP    2089    A
COMCIDP  *              THAT PRECEDE THE STMT KEYWORD.                       COMCIDP    2090    A
COMCIDP  *                                                                    COMCIDP    2091    A
COMCIDP  *           2. SEARCH FOR THE STMT KEYWORD VIA THE LINKED LIST(S)   COMCIDP    2092    A
COMCIDP  *              OF KEYWORD TABLES.                                   COMCIDP    2093    A
COMCIDP  *                                                                    COMCIDP    2094    A
COMCIDP  *           3. TRANSFER CONTROL TO THE APPROPRIATE STMT PROCESSOR   COMCIDP    2095    A
COMCIDP  *              FOR THIS KEYWORD.                                    COMCIDP    2096    A
COMCIDP                                                                       COMCIDP    2097    A
COMCIDP   IDP12    MX0    TB.TOCL                                             COMCIDP    2098    A
COMCIDP            BX1    X0*X5       (X1) = KEYWORD, IN -L- FMT              COMCIDP    2099    A
COMCIDP            ZR     X5,IDP=MN   IF A NULL STMT, IGNORE...               COMCIDP    2100    A
COMCIDP  *                               (PROBABLE "EOS" "EOS" SYNTAX)        COMCIDP    2101    A
COMCIDP            SX6    X5-O.KEY                                            COMCIDP    2102    A
COMCIDP            ZR     X6,IDP13    IF *KEY* TOKEN, NO *STMT CTL*...        COMCIDP    2103    A
COMCIDP                                                                       COMCIDP    2104    A
COMCIDP  *         CHECK FOR *STATEMENT CONTROL* TOKEN.                      COMCIDP    2105    A
COMCIDP                                                                       COMCIDP    2106    A
COMCIDP            BX1    -X0*X5      (X1) = *TOT*                            COMCIDP    2107    A
COMCIDP            SB6    FW.SCT      (B6) = FWA OF *STMT CTL* PROCESSOR TABLE COMCIDP    2108    A
COMCIDP            LX1    SC.TOTP-0                                           COMCIDP    2109    A
COMCIDP            MX0    SC.TOTL     (X0) = EXTRACTION MASK FOR *SKT*        COMCIDP    2110    A
COMCIDP            RJ     SKT         SEARCH IDP TABLE                        COMCIDP    2111    A
COMCIDP            ZR     X2,E.SC     IF NO FIND                              COMCIDP    2112    A
COMCIDP            LX2    0-SC.PADP                                           COMCIDP    2113    A
COMCIDP            SB2    X2+         (B2) = ADDR OF PROCESSOR FOR THIS *TOT* COMCIDP    2114    A
COMCIDP            JP     B2                                                  COMCIDP    2115    A
COMCIDP                                                                       COMCIDP    2116    A
COMCIDP  *         CLASSIFY THIS STATEMENT.                                  COMCIDP    2117    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *                                                                  COMCIDP    2118    A
COMCIDP  *         NOTE THAT *CST* DOES NOT RETURN IF AN ERROR IS DETECTED. COMCIDP    2119    A
COMCIDP                                                                     COMCIDP    2120    A
COMCIDP   IDP13    RJ     CST         CLASSIFY STATEMENT                    COMCIDP    2121    A
COMCIDP            ZR     X6,E.KEY    IF NO FIND, ERROR...                  COMCIDP    2122    A
COMCIDP            SA6    TB=KEY      SAVE ADDR OF FOUND *KEYW* ENTRY       COMCIDP    2123    A
COMCIDP                                                                     COMCIDP    2124    A
COMCIDP  *         HERE IF WE HAVE A KEYWORD MATCH.                         COMCIDP    2125    A
COMCIDP  *                                                                  COMCIDP    2126    A
COMCIDP  *         READY TO TRANSFER CONTROL TO THE APPROPRIATE STMT PROCESSOR. COMCIDP 2127    A
COMCIDP  *         FIRST, HOWEVER, WE NEED TO FIDDLE WITH (A5) AND (X5) SO  COMCIDP    2128    A
COMCIDP  *         THAT UPON ENTRY TO THE STMT PROCESSOR --                 COMCIDP    2129    A
COMCIDP  *                                                                  COMCIDP    2130    A
COMCIDP  *               (A1,X1)= A+C OF *KEYW* ENTRY, WORD 1.              COMCIDP    2131    A
COMCIDP  *               (A5+1) = ADDR OF 1ST TOKEN TO LOOK AT.             COMCIDP    2132    A
COMCIDP  *               (X5)   = .ZR. IF KEYWORD IMMEDIATELY FOLLOWED BY EOS, COMCIDP 2133    A
COMCIDP  *                        ELSE .NZ.                                 COMCIDP    2134    A
COMCIDP                                                                     COMCIDP    2135    A
COMCIDP   IDP14    BX6    X5                                                COMCIDP    2136    A
COMCIDP            SX7    X5-O.COM                                          COMCIDP    2137    A
COMCIDP            ZR     X7,IDP15    IF KEYWORD FOLLOWED BY COMMA          COMCIDP    2138    A
COMCIDP            SA5    A5-B1                                             COMCIDP    2139    A
COMCIDP            LX5    X6                                                COMCIDP    2140    A
COMCIDP                                                                     COMCIDP    2141    A
COMCIDP   IDP15    SA1    TB=KEY      (X1) = ADDR OF *KEYW* ENTRY FOR THIS KEYWRD COMCIDP 2142    A
COMCIDP            SA2    X1+         (A2,X2) = A+C OF *KEYW* ENTRY, WORD 1 COMCIDP    2143    A
COMCIDP            LX2    0-KW.PADP                                         COMCIDP    2144    A
COMCIDP            SB2    X2          (B2) = ADDR OF STMT PROCESSOR         COMCIDP    2145    A
COMCIDP            RJ     CHK         CHECK CM ADDR (BEST TO BE CAREFUL...) COMCIDP    2146    A
COMCIDP            SA2    TB=KEY                                            COMCIDP    2147    A
COMCIDP            MI     B2,SE.PAD   IF STMT PROCESSOR ADDR IS BAD         COMCIDP    2148    A
COMCIDP            SA1    X2          (A1,X1) = A+C OF FOUND *KEYW* ENTRY, WORD 1 COMCIDP 2149    A
COMCIDP            JP     B2          FLY AWAY...                           COMCIDP    2150    A
COMCIDP   IDP=XIT  SPACE  4,10                                               COMCIDP    2151    A
COMCIDP  **        IDP=XIT - EXIT IDP.                                      COMCIDP    2152    A
COMCIDP  *                                                                  COMCIDP    2153    A
COMCIDP  *                                                                  COMCIDP    2154    A
COMCIDP                                                                     COMCIDP    2155    A
COMCIDP   IDP=XIT  BSS    0           ** ENTRY **                           COMCIDP    2156    A
COMCIDP            SA1    IDPPREG     (X1) = ADDR OF NEXT INSTRUCTION TO XEQ COMCIDP   2157    A
COMCIDP            SA2    IDP=JPS     (X2) = *JP B0+0* PLUG SKELETON        COMCIDP    2158    A
COMCIDP            LX1    30                                                COMCIDP    2159    A
COMCIDP            BX6    X2+X1       (X6) = 30/JP B0+PREG,30/NO-OP         COMCIDP    2160    A
COMCIDP            SA6    IDP=JP                                            COMCIDP    2161    A
COMCIDP                                                                     COMCIDP    2162    A
COMCIDP            SX1    FW.SVR      (X1) = FWA OF SAVED REGISTERS         COMCIDP    2163    A
COMCIDP            RJ     RIR         RESTORE INTERACTIVE REGISTERS         COMCIDP    2164    A
COMCIDP                                                                     COMCIDP    2165    A
COMCIDP   IDP=JP   JP     *+4S15      PLUGGED W/ *JP B0+RETURN ADDR*        COMCIDP    2166    A
COMCIDP                                                                     COMCIDP    2167    A
COMCIDP   IDP=JPS  JP     B0+0        PLUG SKELETON                         COMCIDP    2168    A
COMCIDP   IDP=ER   SPACE  4,10                                               COMCIDP    2169    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *         HERE FOR ERROR CONDITION.                                COMCIDP    2170    A
COMCIDP                                                                     COMCIDP    2171    A
COMCIDP   IDP=ER   BSS    0           ** ERROR **                           COMCIDP    2172    A
COMCIDP   IDP.ER   BSS    0                                                 COMCIDP    2173    A
COMCIDP            SX6    -1          SET TO *NO ERROR MSG AVAILABLE*       COMCIDP    2174    A
COMCIDP                                                                     COMCIDP    2175    A
COMCIDP   IDP=ERR  BSS    0           ** ENTRY FROM /ERROR/ MACRO **        COMCIDP    2176    A
COMCIDP            SA6    IDPERR                                            COMCIDP    2177    A
COMCIDP            WRITEC =XF.IDO,(=C. ?.)                                  COMCIDP    2178    A
COMCIDP            EQ     IDP8        TRASH CURRENT LINE/READ NEW LINE...   COMCIDP    2179    A
COMCIDP   IDP=SER  SPACE  4,10                                              COMCIDP    2180    A
COMCIDP  **        IDP=SER - *SYSTEM* ERROR.                                COMCIDP    2181    A
COMCIDP  *                                                                  COMCIDP    2182    A
COMCIDP  *                                                                  COMCIDP    2183    A
COMCIDP  *         HERE WHEN IDP SENSES AN INTERNAL-TYPE ERROR.             COMCIDP    2184    A
COMCIDP                                                                     COMCIDP    2185    A
COMCIDP                                                                     COMCIDP    2186    A
COMCIDP   IDP=SER  BSS    0           ** ENTRY **                           COMCIDP    2187    A
COMCIDP            SA6    IDPERR                                            COMCIDP    2188    A
COMCIDP            SB2    =YF.BDO                                           COMCIDP    2189    A
COMCIDP            RJ     FOF         FLUSH OUTPUT FILE                     COMCIDP    2190    A
COMCIDP            RJ     PEM         PRINT ERROR MSG                       COMCIDP    2191    A
COMCIDP            EQ     IDP8        TRASH CURRENT LINE/READ NEW LINE...   COMCIDP    2192    A
COMCIDP   IDP=IFR  SPACE  4,10                                              COMCIDP    2193    A
COMCIDP  **        IDP=IFR - IDP RESTART.                                   COMCIDP    2194    A
COMCIDP  *                                                                  COMCIDP    2195    A
COMCIDP  *         HERE FROM *RHH* WHEN HOST HAS JUST BEEN RESTARTED        COMCIDP    2196    A
COMCIDP  *         VIA *IFR*.  NEED TO --                                   COMCIDP    2197    A
COMCIDP  *                                                                  COMCIDP    2198    A
COMCIDP  *           1.*MEM* BACK DOWN TO ORIGINAL FIELD LENGTH.            COMCIDP    2199    A
COMCIDP  *                                                                  COMCIDP    2200    A
COMCIDP  *           2. RESTORE THE WORDS AT (RA.SSW), (FL-2), AND (FL-1).  COMCIDP    2201    A
COMCIDP  *                                                                  COMCIDP    2202    A
COMCIDP  *           3. (RE)OPEN AND CONNECT INTERACTIVE FILES, JUST IN CASE COMCIDP    2203    A
COMCIDP  *              USER IS RESTARTING ON ANOTHER DAY.                  COMCIDP    2204    A
COMCIDP  *                                                                  COMCIDP    2205    A
COMCIDP  *           4. CALL *UFR* (USER FREEZE RESTART OWNCODE) TO ALLOW   COMCIDP    2206    A
COMCIDP  *              THE HOST TO PERFORM FREEZE RESTART OWNCODE  (LIKE   COMCIDP    2207    A
COMCIDP  *              RE-SETTING UP FILES, FER INSTANCE...).              COMCIDP    2208    A
COMCIDP                                                                     COMCIDP    2209    A
COMCIDP                                                                     COMCIDP    2210    A
COMCIDP   IDP=IFR  BSS    0           ** ENTRY **                           COMCIDP    2211    A
COMCIDP            SA4    MEM#HOST                                          COMCIDP    2212    A
COMCIDP            SX3    B1                                                COMCIDP    2213    A
COMCIDP            BX6    -X3*X4      CLEAR COMPLETE BIT                    COMCIDP    2214    A
COMCIDP            SA6    A4                                                COMCIDP    2215    A
COMCIDP            MEMORY SCM,A4,RCL                                        COMCIDP    2216    A
COMCIDP                                                                     COMCIDP    2217    A
COMCIDP            SA1    FRZ#SAV                                           COMCIDP    2218    A
COMCIDP            SA2    A1+B1                                             COMCIDP    2219    A
COMCIDP            SA3    A2+B1                                             COMCIDP    2220    A
COMCIDP            LX6    X1                                                COMCIDP    2221    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            BX7    X2                                                  COMCIDP    2222    A
COMCIDP            AX4    30                                                  COMCIDP    2223    A
COMCIDP            =A6    RA.SSW      RESTORE (RA.SSW)                        COMCIDP    2224    A
COMCIDP            SA7    X4-2        RESTORE (FL-2)                          COMCIDP    2225    A
COMCIDP            BX6    X3                                                  COMCIDP    2226    A
COMCIDP            SA6    A7+B1       RESTORE (FL-1)                          COMCIDP    2227    A
COMCIDP            RJ     IIF         INITIALIZE INTERACTIVE FILES            COMCIDP    2228    A
COMCIDP            RJ     UFR         USER FREEZE RESTART OWNCODE             COMCIDP    2229    A
COMCIDP            EQ     IDP=MN      CONTINUE INTERACTIVE SESSION...         COMCIDP    2230    A
COMCIDP   .IDPOS   ENDIF                                                      CCGA018      10    A
COMCIDP   REG      SPACE  4,10                                                COMCIDP    2231    A
COMCIDP  **        REG - REGISTER SNAPSHOT.                                   COMCIDP    2232    A
COMCIDP  *                                                                    COMCIDP    2233    A
COMCIDP  *                                                                    COMCIDP    2234    A
COMCIDP  *                CALLED BY *REG* MACRO.                              COMCIDP    2235    A
COMCIDP  *                                                                    COMCIDP    2236    A
COMCIDP  *         ENTRY  LOWER HALF OF *RJ REG=* WORD HAS FWA OF PARAMETER LIST.  COMCIDP  2237  A
COMCIDP  *                                                                    COMCIDP    2238    A
COMCIDP  * +       RJ     REG=                                                COMCIDP    2239    A
COMCIDP  * -       VFD    30/FWA OF PARAMETER LIST                            COMCIDP    2240    A
COMCIDP  *                                                                    COMCIDP    2241    A
COMCIDP  *         PARAMETER LIST EXISTS IN LOCAL BLOCK *USE DEBUG*, AS SET UP BY  COMCIDP  2242  A
COMCIDP  *         *REG* MACRO (SEE *COMAIDP*) --                             COMCIDP    2243    A
COMCIDP  *                                                                    COMCIDP    2244    A
COMCIDP  *         VFD    60/LOWER LIMIT (LL)                                 COMCIDP    2245    A
COMCIDP  *         VFD    60/UPPER LIMIT (UL)                                 COMCIDP    2246    A
COMCIDP  *         VFD    60/INCREMENT (INC)                                  COMCIDP    2247    A
COMCIDP  *         VFD    60/10HNAME  (OR .ZR. IF NO NAME)                    COMCIDP    2248    A
COMCIDP  *         VFD    60/0        (USED BY *FRK* TO KEEP SNAP COUNT)      COMCIDP    2249    A
COMCIDP  *         VFD    60/USER FLAGS                                       COMCIDP    2250    A
COMCIDP  *         VFD    60/REGISTER DUMP FLAG                               COMCIDP    2251    A
COMCIDP  *                .                                                   COMCIDP    2252    A
COMCIDP  *                .                                                   COMCIDP    2253    A
COMCIDP  *         VFD    60/REGISTER DUMP FLAG                               COMCIDP    2254    A
COMCIDP  *                .ZR. = NO REGISTER DUMP  (OR END OF REGISTER LIST)  COMCIDP    2255    A
COMCIDP  *                .MI. = DUMP ALL REGISTERS                           COMCIDP    2256    A
COMCIDP  *                .GT.0 = ADDR OF REGISTER SAVE WORD FOR REGISTER TO DMP  COMCIDP  2257  A
COMCIDP  *                                                                    COMCIDP    2258    A
COMCIDP  *         EXIT   NONE                                                COMCIDP    2259    A
COMCIDP  *                                                                    COMCIDP    2260    A
COMCIDP  *         USES   NONE        (ALL REGISTERS ARE SAVED AND RESTORED)  COMCIDP    2261    A
COMCIDP  *                                                                    COMCIDP    2262    A
COMCIDP  *         CALLS  DAR,DSR,FRK,HDR,RSR=,SOB,SVR=,URO                   COMCIDP    2263    A
COMCIDP                                                                       COMCIDP    2264    A
COMCIDP                                                                       COMCIDP    2265    A
COMCIDP   REG=     SUBR               ** ENTRY/EXIT **                        COMCIDP    2266    A
COMCIDP   +        RJ     =XSVR       SAVE ALL REGISTERS                      COMCIDP    2267    A
COMCIDP   -        VFD    30/FW.SVR                                           COMCIDP    2268    A
COMCIDP            SX1    1           SET TO *EXECUTIVE IS /REG=/*            COMCIDP    2269    A
COMCIDP            RJ     SOB         SET OUTPUT BIT FLAGS                    COMCIDP    2270    A
COMCIDP            SA1    REG=        (X1) = 30/EQ CALLING ADDRESS+1, 30/0    COMCIDP    2271    A
COMCIDP            RJ     IEX         INITIALIZE EXECUTIVE                    COMCIDP    2272    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            RJ     URO         USER REG= OWNCODE                                  COMCIDP    2273    A
COMCIDP            PL     X1,REG3     IF NO REGISTER SNAPSHOT THIS TIME                  COMCIDP    2274    A
COMCIDP            RJ     FRK         CHECK FREQUENCY PARAMETERS                         COMCIDP    2275    A
COMCIDP            ZR     X5,REG3     IF NOT SNAPPING THIS TIME                          COMCIDP    2276    A
COMCIDP            RJ     HDR         OUTPUT SNAP HEADER                                 COMCIDP    2277    A
COMCIDP            SA5    FW=PARM                                                        COMCIDP    2278    A
COMCIDP            SA1    X5+SN=RRL   (A1,X1) = ADDR + CNTS OF 1ST REGISTER DUMP         COMCIDP    2279    A
COMCIDP  *                               PARAMETER WORD                                  COMCIDP    2280    A
COMCIDP            PL     X1,REG2     IF NOT DUMPING ALL REGISTERS                       COMCIDP    2281    A
COMCIDP            RJ     DAR         DUMP ALL REGISTERS                                 COMCIDP    2282    A
COMCIDP            EQ     REG3                                                           COMCIDP    2283    A
COMCIDP                                                                                   COMCIDP    2284    A
COMCIDP   REG2     RJ     DSR         DUMP SELECTED REGISTERS                            COMCIDP    2285    A
COMCIDP                                                                                   COMCIDP    2286    A
COMCIDP   REG3     SX1    FW.SVR      (X1) = FWA OF SAVED REGISTERS                      COMCIDP    2287    A
COMCIDP            RJ     RIR         RESTORE INTERACTIVE REGISTERS                      COMCIDP    2288    A
COMCIDP            EQ     EXIT.                                                          COMCIDP    2289    A
COMCIDP   SNP      SPACE  4,10                                                           COMCIDP    2290    A
COMCIDP  **        SNP - SNAPSHOT OF CORE AND REGISTERS.                                 COMCIDP    2291    A
COMCIDP  *                                                                               COMCIDP    2292    A
COMCIDP  *                                                                               COMCIDP    2293    A
COMCIDP  *                 CALLED BY *SNAP* MACRO.                                       COMCIDP    2294    A
COMCIDP  *                                                                               COMCIDP    2295    A
COMCIDP  *         LOWER HALF OF *RJ SNP=* WORD HAS FWA OF PARAMETER LIST--              COMCIDP    2296    A
COMCIDP  *                                                                               COMCIDP    2297    A
COMCIDP  * +       RJ     SNP=                                                           COMCIDP    2298    A
COMCIDP  * -       VFD    30/FWA OF PARAMETER LIST                                       COMCIDP    2299    A
COMCIDP  *                                                                               COMCIDP    2300    A
COMCIDP  *         PARAMETER LIST EXISTS IN LOCAL BLOCK *USE DEBUG*, AS SET UP BY        COMCIDP    2301    A
COMCIDP  *         BY *SNAP* MACRO. (SEE *COMAIDP*) --                                   COMCIDP    2302    A
COMCIDP  *                                                                               COMCIDP    2303    A
COMCIDP  *         VFD    60/LOWER LIMIT (LL)                                            COMCIDP    2304    A
COMCIDP  *         VFD    60/UPPER LIMIT (UL)                                            COMCIDP    2305    A
COMCIDP  *         VFD    60/INCREMENT (INC)                                             COMCIDP    2306    A
COMCIDP  *         VFD    60/10HNAME  (OR .ZR. IF NO NAME)                               COMCIDP    2307    A
COMCIDP  *         VFD    60/0        (USED BY *FRK* TO KEEP SNAP COUNT)                 COMCIDP    2308    A
COMCIDP  *         VFD    18/0,21/LVL OF INDIRECT ADDRESSING,21/FWA                      COMCIDP    2309    A
COMCIDP  *         VFD    18/0,21/LVL OF INDIRECT ADDRESSING,21/LWA                      COMCIDP    2310    A
COMCIDP  *         VFD    18/0,21/LVL OF INDIRECT ADDRESSING,21/LEN                      COMCIDP    2311    A
COMCIDP  *         VFD    60/USER FLAGS                                                  COMCIDP    2312    A
COMCIDP  *         VFD    60/REGISTER DUMP FLAG                                          COMCIDP    2313    A
COMCIDP  *                 .                                                             COMCIDP    2314    A
COMCIDP  *                 .                                                             COMCIDP    2315    A
COMCIDP  *         VFD    60/REGISTER DUMP FLAG                                          COMCIDP    2316    A
COMCIDP  *                .ZR. = NO REGISTER DUMP  (OR END OF REGISTER LIST)             COMCIDP    2317    A
COMCIDP  *                .MI. = DUMP ALL REGISTERS                                      COMCIDP    2318    A
COMCIDP  *                .GT.0 = ADDR OF REGISTER SAVE WORD FOR REGISTER TO DMP         COMCIDP    2319    A
COMCIDP  *                                                                               COMCIDP    2320    A
COMCIDP  *         USES   NONE        (ALL REGISTERS ARE SAVED AND RESTORED)             COMCIDP    2321    A
COMCIDP  *                                                                               COMCIDP    2322    A
COMCIDP  *         CALLS  DAR,DCM,DSR,FRK,HDR,RIR,SOB,SVR,USO                            COMCIDP    2323    A
COMCIDP                                                                                  COMCIDP    2324    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP                                                                                  COMCIDP     2325     A
COMCIDP   SNP=     SUBR                 ** ENTRY/EXIT **                                 COMCIDP     2326     A
COMCIDP   +        RJ     =XSVR         SAVE ALL REGISTERS                               COMCIDP     2327     A
COMCIDP   -        VFD    30/FW.SVR                                                      COMCIDP     2328     A
COMCIDP            SX1    2             SET TO *EXECUTIVE IS /SNP=/*                     COMCIDP     2329     A
COMCIDP            RJ     SOB           SET OUTPUT BIT FLAGS                             COMCIDP     2330     A
COMCIDP            SA1    SNP=          (X1) = 30/EQ CALLING ADDRESS+1, 30/0             COMCIDP     2331     A
COMCIDP            RJ     IEX           INITIALIZE EXECUTIVE                             COMCIDP     2332     A
COMCIDP            RJ     USO           USER SNP= OWNCODE                                COMCIDP     2333     A
COMCIDP            PL     X1,SNP4       IF NO SNAPSHOT THIS TIME                         COMCIDP     2334     A
COMCIDP            RJ     FRK           CHECK FREQUENCY PARAMETERS                       COMCIDP     2335     A
COMCIDP            ZR     X5,SNP4       IF NO SNAP THIS TIME                             COMCIDP     2336     A
COMCIDP            RJ     HDR           OUTPUT SNAP HEADER                               COMCIDP     2337     A
COMCIDP            SA1    X5+SN=SRL     (A1,X1) = ADDR + CNTS OF 1ST WORD OF             COMCIDP     2338     A
COMCIDP   *                                REGISTER LIST                                COMCIDP     2339     A
COMCIDP            PL     X1,SNP2       IF NOT DUMPING ALL REGISTERS                     COMCIDP     2340     A
COMCIDP            RJ     DAR           DUMP ALL REGISTERS                               COMCIDP     2341     A
COMCIDP            EQ     SNP3                                                           COMCIDP     2342     A
COMCIDP                                                                                  COMCIDP     2343     A
COMCIDP   SNP2     RJ     DSR           DUMP SELECTED REGISTERS                          COMCIDP     2344     A
COMCIDP                                                                                  COMCIDP     2345     A
COMCIDP   SNP3     SA5    FW=PARM                                                        COMCIDP     2346     A
COMCIDP            SA2    X5+SN=FWA     (X2) = 10/0,21/LVL OF IND ADDR,21/FWA            COMCIDP     2347     A
COMCIDP            SA3    A2+B1         (X3) = 18/0,21/LVL OF IND ADDR,21/LWA            COMCIDP     2348     A
COMCIDP            SA4    A3+B1         (X4) = 18/0,21/LVL OF IND ADDR,21/LEN            COMCIDP     2349     A
COMCIDP            RJ     DCM           DUMP CENTRAL MEMORY                              COMCIDP     2350     A
COMCIDP                                                                                  COMCIDP     2351     A
COMCIDP   SNP4     SX1    FW.SVR        (X1) = FWA OF SAVED REGISTERS                    COMCIDP     2352     A
COMCIDP            RJ     RIR           RESTORE INTERACTIVE REGISTERS                    COMCIDP     2353     A
COMCIDP            EQ     EXIT.                                                          COMCIDP     2354     A
COMCIDP   IDP      TITLE  IDP "IDPVER"/THE STATEMENT PROCESSORS.                         COMCIDP     2355     A
COMCIDP   .IDPOS   IFNE   .OS,2                                                          CCGA018       11     A
COMCIDP   SC=      SPACE  4,10                                                           COMCIDP     2356     A
COMCIDP  ***       SC= - STATEMENT CONTROL PROCESSORS.                                   COMCIDP     2357     A
COMCIDP  *                                                                               COMCIDP     2358     A
COMCIDP  *                                                                               COMCIDP     2359     A
COMCIDP  *         THE FOLLOWING SECTION CONTAINS THE PROCESSORS THAT HANDLE             COMCIDP     2360     A
COMCIDP  *         THE SO CALLED *STATEMENT CONTROL* SYNTAXS.  THESE ARE THE             COMCIDP     2361     A
COMCIDP  *         SPECIAL PREFIX TOKENS THAT CAN OPTIONALLY PRECEDE ANY IDP             COMCIDP     2362     A
COMCIDP  *         COMMAND STMT, AND THAT CONTROL HOW THAT STMT IS **SEEN**.             COMCIDP     2363     A
COMCIDP  *                                                                               COMCIDP     2364     A
COMCIDP  *         FOR EXAMPLE, THE *BRIEF* MODE TOGGLE SWITCH, *.*, IS NOT              COMCIDP     2365     A
COMCIDP  *         REALLY A KEYWORD IN THE STRUCTURAL SENSE.  IT MERELY INDICATES        COMCIDP     2366     A
COMCIDP  *         TO IDP THAT THE USER WISHES TO TOGGLE *BRIEF* MODE.                   COMCIDP     2367     A
COMCIDP   SC=BRF   SPACE  4,10                                                           COMCIDP     2368     A
COMCIDP  **        HERE TO PROCESS *.* (BRIEF MODE TOGGLE SWITCH).                       COMCIDP     2369     A
COMCIDP                                                                                  COMCIDP     2370     A
COMCIDP   SC=BRF   BSS    0             ** ENTRY **                                      COMCIDP     2371     A
COMCIDP            SA1    IDPFLG                                                         COMCIDP     2372     A
COMCIDP            MX2    1                                                              COMCIDP     2373     A
COMCIDP            LX2    IDF.BRFP-59                                                    COMCIDP     2374     A
COMCIDP            BX6    X1-X2         TOGGLE *BRF*                                     COMCIDP     2375     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            SA6    A1                                                    COMCIDP    2376    A
COMCIDP            EQ     IDP=SC                                                COMCIDP    2377    A
COMCIDP   SC=QQQ   SPACE  4,10                                                  COMCIDP    2378    A
COMCIDP  **        HERE TO PROCESS *?*.                                         COMCIDP    2379    A
COMCIDP                                                                         COMCIDP    2380    A
COMCIDP                                                                         COMCIDP    2381    A
COMCIDP   SC=QQQ   BSS    0           ** ENTRY **                               COMCIDP    2382    A
COMCIDP            RJ     PEM         PRINT ERROR MSG                           COMCIDP    2383    A
COMCIDP            EQ     IDP=SC                                                COMCIDP    2384    A
COMCIDP   ST=      SPACE  4,10                                                  COMCIDP    2385    A
COMCIDP  **        ST= - STATEMENT PROCESSORS.                                  COMCIDP    2386    A
COMCIDP  *                                                                      COMCIDP    2387    A
COMCIDP   ST=ABS   SPACE  4,10                                                  COMCIDP    2388    A
COMCIDP  **        HERE TO PROCESS *ABS*.                                       COMCIDP    2389    A
COMCIDP                                                                         COMCIDP    2390    A
COMCIDP                                                                         COMCIDP    2391    A
COMCIDP   ST=ABS   BSS    0           ** ENTRY **                               COMCIDP    2392    A
COMCIDP            SA1    IDPFLG                                                COMCIDP    2393    A
COMCIDP            MX2    1                                                     COMCIDP    2394    A
COMCIDP            LX2    IDF.ADRP-59                                           COMCIDP    2395    A
COMCIDP            BX6    -X2*X1      CLEAR *ADR*                               COMCIDP    2396    A
COMCIDP            SA6    A1                                                    COMCIDP    2397    A
COMCIDP            EQ     IDP=MN                                                COMCIDP    2398    A
COMCIDP   ST=BRK   SPACE  4,10                                                  COMCIDP    2399    A
COMCIDP  *         HERE TO PROCESS *BREAK ADDR,LL,UL,INC*,                      COMCIDP    2400    A
COMCIDP  *                     OR *BRPL ADDR,ADDR1,LL,UL,INC* FORM,             COMCIDP    2401    A
COMCIDP  *                     OR  *BREQ ADDR,ADDR1,ADDR2,LL,UL,INC* FORM.      COMCIDP    2402    A
COMCIDP                                                                         COMCIDP    2403    A
COMCIDP   ST=BRK   BSS    0           ** ENTRY **                               COMCIDP    2404    A
COMCIDP            LX1    0-KW.XATP                                             COMCIDP    2405    A
COMCIDP            MX2    -KW.XATL                                              COMCIDP    2406    A
COMCIDP            BX6    -X2*X1      (X6) = BREAK TYPE CODE (BTC)              COMCIDP    2407    A
COMCIDP            SA6    IDPBTC                                                COMCIDP    2408    A
COMCIDP            ZR     X5,ST=BRK13 IF EOS ENCOUNTERED, LIST ALL BREAKS...    COMCIDP    2409    A
COMCIDP            RJ     PAS         PARSE SUBEXPRESSION-- ADDR                COMCIDP    2410    A
COMCIDP            ZR     X3,ST=BRK13 IF 1ST EXPRESSION IS NULL, LIST ALL BREAKS COMCIDP   2411    A
COMCIDP            SX6    X1                                                    COMCIDP    2412    A
COMCIDP            SB2    X1          (B2) = ADDR AT WHICH TO BREAK             COMCIDP    2413    A
COMCIDP            SA6    AP=FWA                                                COMCIDP    2414    A
COMCIDP            RJ     CHK         CHECK CM ADDRESS                          COMCIDP    2415    A
COMCIDP            MI     B2,E.BRAD   IF ADDR IS BAD                            COMCIDP    2416    A
COMCIDP            SA2    IDPBTC                                                COMCIDP    2417    A
COMCIDP            MX3    0                                                     COMCIDP    2418    A
COMCIDP            ZR     X2,ST=BRK2  IF THIS IS *BREAK ADDR,LL,UL,INC*         COMCIDP    2419    A
COMCIDP                                                                         COMCIDP    2420    A
COMCIDP  *         ASSEMBLE ADDR1.                                              COMCIDP    2421    A
COMCIDP                                                                         COMCIDP    2422    A
COMCIDP            ZR     X5,ST=BRK2  IF ADDR FOLLOWED BY EOS                   COMCIDP    2423    A
COMCIDP            RJ     PAS         PARSE SUBEXPRESSION-- ADDR1               COMCIDP    2424    A
COMCIDP                                                                         COMCIDP    2425    A
COMCIDP   ST=BRK2  NZ     X3,ST=BRK3  IF ADDR1 NOT NULL                         COMCIDP    2426    A
COMCIDP            SX1    0           (X1) = DEFAULT ADDR1                      COMCIDP    2427    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP                                                                                COMCIDP     2428     A
COMCIDP   ST=BRK3  SB2    X1                                                           COMCIDP     2429     A
COMCIDP            RJ     CHK         CHECK CM ADDRESS                                COMCIDP     2430     A
COMCIDP            MI     B2,E.BRA1   IF ADDR1 IS BAD                                 COMCIDP     2431     A
COMCIDP            SX6    B2                                                          COMCIDP     2432     A
COMCIDP            MX3    0                                                           COMCIDP     2433     A
COMCIDP            SA6    AP=LWA                                                      COMCIDP     2434     A
COMCIDP            SA2    IDPBTC                                                      COMCIDP     2435     A
COMCIDP            SX7    X2-BTC.EQ                                                   COMCIDP     2436     A
COMCIDP            MI     X7,ST=BRK4  IF THIS IS *BREAK ADDR,LL,UL,INC*               COMCIDP     2437     A
COMCIDP   *                              OR *BRPL ADDR,ADDR1,LL,UL,INC* FORM           COMCIDP     2438     A
COMCIDP                                                                                COMCIDP     2439     A
COMCIDP   *        ASSEMBL ADDR2.                                                      COMCIDP     2440     A
COMCIDP                                                                                COMCIDP     2441     A
COMCIDP            ZR     X5,ST=BRK4  IF ADDR1 FOLLOWED BY EOS                        COMCIDP     2442     A
COMCIDP            RJ     PAS         PARSE SUBEXPRESSION-- ADDR2                     COMCIDP     2443     A
COMCIDP                                                                                COMCIDP     2444     A
COMCIDP   ST=BRK4  NZ     X3,ST=BRK5  IF ADDR2 NOT NULL                               COMCIDP     2445     A
COMCIDP            SX1    0                                                           COMCIDP     2446     A
COMCIDP                                                                                COMCIDP     2447     A
COMCIDP   ST=BRK5  SB2    X1                                                           COMCIDP     2448     A
COMCIDP            RJ     CHK         CHECK CM ADDRESS                                COMCIDP     2449     A
COMCIDP            MI     B2,E.BRA2   IF ADDR2 IS BAD                                 COMCIDP     2450     A
COMCIDP            SX6    B2                                                          COMCIDP     2451     A
COMCIDP            MX3    0                                                           COMCIDP     2452     A
COMCIDP            SA6    AP=LEN      SAVE (X6) = ADDR2                               COMCIDP     2453     A
COMCIDP                                                                                COMCIDP     2454     A
COMCIDP   *        ASSEMBLE LL.                                                        COMCIDP     2455     A
COMCIDP                                                                                COMCIDP     2456     A
COMCIDP            ZR     X5,ST=BRK6  IF ADDR2 FOLLOWED BY EOS                        COMCIDP     2457     A
COMCIDP            RJ     PAS         PARSE SUBEXPRESSION-- LL                        COMCIDP     2458     A
COMCIDP                                                                                COMCIDP     2459     A
COMCIDP   ST=BRK6  NZ     X3,ST=BRK7  IF LL NOT NULL                                  COMCIDP     2460     A
COMCIDP            SX1    1           (X1) = DEFAULT LL                               COMCIDP     2461     A
COMCIDP                                                                                COMCIDP     2462     A
COMCIDP   ST=BRK7  SX6    X1                                                           COMCIDP     2463     A
COMCIDP            SA6    AP=LL                                                       COMCIDP     2464     A
COMCIDP            MX3    0                                                           COMCIDP     2465     A
COMCIDP            ZR     X5,ST=BRK8  IF LL FOLLOWED BY EOS                           COMCIDP     2466     A
COMCIDP            RJ     PAS         PARSE SUBEXPRESSION-- UL                        COMCIDP     2467     A
COMCIDP                                                                                COMCIDP     2468     A
COMCIDP   ST=BRK8  NZ     X3,ST=BRK9  IF UL NOT NULL                                  COMCIDP     2469     A
COMCIDP            SX1    100         (X1) = DEFAULT UL                               COMCIDP     2470     A
COMCIDP                                                                                COMCIDP     2471     A
COMCIDP   ST=BRK9  SX6    X1                                                           COMCIDP     2472     A
COMCIDP            SA6    AP=UL                                                       COMCIDP     2473     A
COMCIDP            MX3    0                                                           COMCIDP     2474     A
COMCIDP            ZR     X5,ST=BRK10 IF UL FOLLOWED BY EOS                           COMCIDP     2475     A
COMCIDP            RJ     PAS         PARSE SUBEXPRESSION-- INC                       COMCIDP     2476     A
COMCIDP                                                                                COMCIDP     2477     A
COMCIDP   ST=BRK10 NZ     X3,ST=BRK11 IF INC NOT NULL                                 COMCIDP     2478     A
COMCIDP            SX1    1           (X1) = DEFAULT INC                              COMCIDP     2479     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP                                                                    COMCIDP    2480    A
COMCIDP   ST=BRK11 SX6    X1                                               COMCIDP    2481    A
COMCIDP            SA6    AP=INC                                           COMCIDP    2482    A
COMCIDP                                                                    COMCIDP    2483    A
COMCIDP  *         ASSEMBLE NEW BREAK TABLE ENTRY.                         COMCIDP    2484    A
COMCIDP                                                                    COMCIDP    2485    A
COMCIDP            SA1    AP=FWA      (X1) = BREAK ADDR                    COMCIDP    2486    A
COMCIDP            SA2    A1+B1       (X2) = ADDR1                         COMCIDP    2487    A
COMCIDP            SA3    A2+B1       (X3) = ADDR2                         COMCIDP    2488    A
COMCIDP            SA4    IDPBTC      (X4) = BREAK TYPE CODE               COMCIDP    2489    A
COMCIDP            LX2    54-18                                            COMCIDP    2490    A
COMCIDP            BX6    X1+X2       (X6) = 6/0,18/ADDR1,18/0,18/BREAK ADDR  COMCIDP    2491    A
COMCIDP            LX3    36-18                                            COMCIDP    2492    A
COMCIDP            BX6    X6+X3       (X6) = 6/0,18/ADDR1,18/ADDR2,18/BRK ADDR  COMCIDP    2493    A
COMCIDP            LX4    60-6                                             COMCIDP    2494    A
COMCIDP            BX6    X6+X4       (X6) = 6/BTC,18/ADDR1,18/ADDR2,18/BRK ADDR  COMCIDP    2495    A
COMCIDP            SA6    ST=BRKB                                          COMCIDP    2496    A
COMCIDP                                                                    COMCIDP    2497    A
COMCIDP            SA2    AP=LL       (X2) = LL                            COMCIDP    2498    A
COMCIDP            SA3    A2+B1       (X3) = UL                            COMCIDP    2499    A
COMCIDP            SA4    A3+B1       (X4) = INC                           COMCIDP    2500    A
COMCIDP            LX2    60-15                                            COMCIDP    2501    A
COMCIDP            LX3    45-15                                            COMCIDP    2502    A
COMCIDP            BX7    X2+X3       (X7) = 15/LL,15/UL,12/0,18/0         COMCIDP    2503    A
COMCIDP            LX4    30-12                                            COMCIDP    2504    A
COMCIDP            BX7    X7+X4       (X7) = 15/LL,15/UL,12/INC,18/0       COMCIDP    2505    A
COMCIDP            SA7    A6+B1                                            COMCIDP    2506    A
COMCIDP                                                                    COMCIDP    2507    A
COMCIDP            SX1    X1                                               COMCIDP    2508    A
COMCIDP            SB6    IDPBA                                            COMCIDP    2509    A
COMCIDP            MX0    -18                                              COMCIDP    2510    A
COMCIDP            BX0    -X0                                              COMCIDP    2511    A
COMCIDP            RJ     SKT         SEARCH FOR BREAK ADDR                COMCIDP    2512    A
COMCIDP            SA1    ST=BRKB                                          COMCIDP    2513    A
COMCIDP            NZ     X2,ST=BRK12 IF BREAK ADDR ALREADY IN TABLE       COMCIDP    2514    A
COMCIDP                                                                    COMCIDP    2515    A
COMCIDP  *         HERE IF MAKING A NEW ENTRY.                             COMCIDP    2516    A
COMCIDP                                                                    COMCIDP    2517    A
COMCIDP            BX6    X1                                               COMCIDP    2518    A
COMCIDP            RJ     ADZ         ADD NEW *IDPBA* ENTRY                COMCIDP    2519    A
COMCIDP            SA3    X1          (X3) = PREVIOUS CONTENTS OF BREAK ADDR  COMCIDP    2520    A
COMCIDP            SA4    ST=BRKA     (X4) = *RJ IDP* PLUG                 COMCIDP    2521    A
COMCIDP            BX6    X3                                               COMCIDP    2522    A
COMCIDP            LX7    X4                                               COMCIDP    2523    A
COMCIDP            SB6    B7+B7                                            COMCIDP    2524    A
COMCIDP            SA6    IDPBC+B6                                         COMCIDP    2525    A
COMCIDP            SA7    A3                                               COMCIDP    2526    A
COMCIDP            SA1    ST=BRKB+1   (X1) = 2ND WORD OF NEW *IDPBC* ENTRY COMCIDP    2527    A
COMCIDP            BX6    X1                                               COMCIDP    2528    A
COMCIDP            SA6    A6+B1                                            COMCIDP    2529    A
COMCIDP            EQ     IDP=MN                                           COMCIDP    2530    A
COMCIDP                                                                    COMCIDP    2531    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *         HERE IF REPLACING/UPDATING AN EXISTING ENTRY.                    COMCIDP    2532    A
COMCIDP                                                                             COMCIDP    2533    A
COMCIDP   ST=BRK12 SB6    B7+B7                                                     COMCIDP    2534    A
COMCIDP            BX6    X1                                                        COMCIDP    2535    A
COMCIDP            SA3    IDPBC+B6+1  (X3) = 2ND WORD OF EXISTING *IDPBC* ENTRY     COMCIDP    2536    A
COMCIDP            SA6    A2          REPLACE OLD *IDPBA* WITH NEW                  COMCIDP    2537    A
COMCIDP            SA4    A1+B1       (X4) = NEW 2ND WORD OF *IDPBC* ENTRY          COMCIDP    2538    A
COMCIDP            SX3    X3          (X3) = SNAP COUNT                             COMCIDP    2539    A
COMCIDP            BX6    X4+X3                                                     COMCIDP    2540    A
COMCIDP            SA6    A3          REPLACE OLD WITH NEW                          COMCIDP    2541    A
COMCIDP            EQ     IDP=MN                                                    COMCIDP    2542    A
COMCIDP                                                                             COMCIDP    2543    A
COMCIDP  *         HERE TO LIST ALL BREAKS.                                         COMCIDP    2544    A
COMCIDP                                                                             COMCIDP    2545    A
COMCIDP   ST=BRK13 RJ     LBT         LIST BREAK TABLE                              COMCIDP    2546    A
COMCIDP            EQ     IDP=MN                                                    COMCIDP    2547    A
COMCIDP                                                                             COMCIDP    2548    A
COMCIDP                                                                             COMCIDP    2549    A
COMCIDP   ST=BRKA  RJ     IDP=        PLUG VALUE                                    COMCIDP    2550    A
COMCIDP   -        VFD    30/0                                                      COMCIDP    2551    A
COMCIDP                                                                             COMCIDP    2552    A
COMCIDP   ST=BRKB  BSSZ   2           SAVE CELLS                                    COMCIDP    2553    A
COMCIDP   ST=COD   SPACE  4,10                                                      COMCIDP    2554    A
COMCIDP  *         HERE TO PROCESS *CODE FWA,LWA,LEN*.                              COMCIDP    2555    A
COMCIDP                                                                             COMCIDP    2556    A
COMCIDP   ST=COD   BSS    0                                                         COMCIDP    2557    A
COMCIDP            RJ     PAT         PARSE *FWA,LWA,LEN* TRIPLE                    COMCIDP    2558    A
COMCIDP            MI     B2,E.FLL    IF FWA, LWA, OR LEN IS BAD                    COMCIDP    2559    A
COMCIDP                                                                             COMCIDP    2560    A
COMCIDP  *         HERE TO DEASSEMBLE FROM FWA TO LWA.                              COMCIDP    2561    A
COMCIDP                                                                             COMCIDP    2562    A
COMCIDP   ST=COD5  SA1    AP=FWA                                                    COMCIDP    2563    A
COMCIDP            SA2    A1+B1                                                     COMCIDP    2564    A
COMCIDP            IX3    X2-X1                                                     COMCIDP    2565    A
COMCIDP            SX6    X1+B1                                                     COMCIDP    2566    A
COMCIDP            MI     X3,IDP=MN   IF FINISHED                                   COMCIDP    2567    A
COMCIDP            SA6    A1                                                        COMCIDP    2568    A
COMCIDP            SA5    X1          (A5,X5) = WORD TO DEASSEMBLE                  COMCIDP    2569    A
COMCIDP            SB4    60          (B4) = POSITION COUNTER                       COMCIDP    2570    A
COMCIDP                                                                             COMCIDP    2571    A
COMCIDP   ST=COD6  RJ     DAZ         DEASSEMBLE INSTRUCTION                        COMCIDP    2572    A
COMCIDP            MI     X1,ST=COD5  IF INSTRUCTION FORCED UPPER                   COMCIDP    2573    A
COMCIDP            SB4    B4-B5                                                     COMCIDP    2574    A
COMCIDP            GT     B4,B0,ST=COD6  IF MORE INSTRUCTIONS TO GO IN THIS WORD    COMCIDP    2575    A
COMCIDP            EQ     ST=COD5                                                   COMCIDP    2576    A
COMCIDP   ST=CON   SPACE  4,10                                                      COMCIDP    2577    A
COMCIDP  *         HERE TO PROCESS *CONNECT FILE*.                                  COMCIDP    2578    A
COMCIDP                                                                             COMCIDP    2579    A
COMCIDP   ST=CON   ZR     X5,E.LFN    IF EOS ENCOUNTERED                            COMCIDP    2580    A
COMCIDP            SA5    A5+B1                                                     COMCIDP    2581    A
COMCIDP            MX2    TB.TOCL                                                   COMCIDP    2582    A
COMCIDP            BX1    X2*X5       (X1) = LFN, -L- FMT                           COMCIDP    2583    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            RJ     SSY         SEARCH SYMBOL TABLES                          COMCIDP    2584    A
COMCIDP            ZR     B6,E.VAR    IF NO FIND                                    COMCIDP    2585    A
COMCIDP            SB2    X6+         (B2) = FWA OF FET                             COMCIDP    2586    A
COMCIDP            RJ     CHK         CHECK CM ADDR                                 COMCIDP    2587    A
COMCIDP            MI     B2,E.ADDR   IF ADDR IS BAD                                COMCIDP    2588    A
COMCIDP            SA1    B2          (A1,X1) = A+C OF WORD 1 OF FET                COMCIDP    2589    A
COMCIDP            SX2    B0          SET TO *CONNECT*                              COMCIDP    2590    A
COMCIDP            RJ     CON         CONNECT/DISCONT FILE                          COMCIDP    2591    A
COMCIDP                                                                             COMCIDP    2592    A
COMCIDP   #OS3     IFEQ   .OS,3       IF SCOPE 3                                    COMCIDP    2593    A
COMCIDP            MI     X1,SE.CON   IF *CON* DETECTED ERROR...                    COMCIDP    2594    A
COMCIDP   #OS3     ENDIF                                                            COMCIDP    2595    A
COMCIDP   #UFT     ENDIF                                                            COMCIDP    2596    A
COMCIDP                                                                             COMCIDP    2597    A
COMCIDP            EQ     IDP=MN                                                    COMCIDP    2598    A
COMCIDP   ST=DPC   SPACE  4,10                                                      COMCIDP    2599    A
COMCIDP  *         HERE TO PROCESS *DPC FWA,LWA,LEN*.                               COMCIDP    2600    A
COMCIDP                                                                             COMCIDP    2601    A
COMCIDP   ST=DPC   BSS    0                                                         COMCIDP    2602    A
COMCIDP            RJ     PAT         PARSE *FWA,LWA,LEN* TRIPLE                    COMCIDP    2603    A
COMCIDP            MI     B2,E.FLL    IF FWA, LWA, OR LEN IS BAD                    COMCIDP    2604    A
COMCIDP            RJ     DOD         DUMP CENTRAL MEMORY - OCTAL AND DPC           COMCIDP    2605    A
COMCIDP            PL     B2,IDP=MN   IF FWA,LWA,AND LEN OK                         COMCIDP    2606    A
COMCIDP            EQ     SE.FLL                                                    COMCIDP    2607    A
COMCIDP   ST=DSC   SPACE  4,10                                                      COMCIDP    2608    A
COMCIDP  *         HERE TO PROCESS *DISCONT FILE*.                                  COMCIDP    2609    A
COMCIDP                                                                             COMCIDP    2610    A
COMCIDP   ST=DSC   ZR     X5,E.LFN    IF EOS ENCOUNTERED                            COMCIDP    2611    A
COMCIDP            SA5    A5+B1                                                     COMCIDP    2612    A
COMCIDP            MX2    TB.TOCL                                                   COMCIDP    2613    A
COMCIDP            BX1    X2*X5       (X1) = LFN, -L- FMT                           COMCIDP    2614    A
COMCIDP            RJ     SSY         SEARCH SYMBOL TABLES                          COMCIDP    2615    A
COMCIDP            ZR     B6,E.VAR    IF NO FIND                                    COMCIDP    2616    A
COMCIDP            SB2    X6+         (B2) = FWA OF FET                             COMCIDP    2617    A
COMCIDP            RJ     CHK         CHECK CM ADDR                                 COMCIDP    2618    A
COMCIDP            MI     B2,E.ADDR   IF ADDR IS BAD                                COMCIDP    2619    A
COMCIDP            SA1    B2          (A1,X1) = A+C OF WORD 1 OF FET                COMCIDP    2620    A
COMCIDP            SB2    B1          SET TO *DISCONT*                              COMCIDP    2621    A
COMCIDP            RJ     CON         CONNECT/DISCONT FILE                          COMCIDP    2622    A
COMCIDP                                                                             COMCIDP    2623    A
COMCIDP            IFEQ   .OS,3,1     IF SCOPE 3                                    COMCIDP    2624    A
COMCIDP            MI     X1,SE.CON   IF *CON* DETECTED ERROR...                    COMCIDP    2625    A
COMCIDP                                                                             COMCIDP    2626    A
COMCIDP            EQ     IDP=MN                                                    COMCIDP    2627    A
COMCIDP   ST=END   SPACE  4,10                                                      COMCIDP    2628    A
COMCIDP  **        HERE TO PROCESS *END*.                                           COMCIDP    2629    A
COMCIDP  *                                                                          COMCIDP    2630    A
COMCIDP  *           1. ISSUE THE FAREWELL PROMPT, *BYE...*, AND FLUSH              COMCIDP    2631    A
COMCIDP  *              THE INTERACTIVE OUTPUT LFN, *F.IDO*.                        COMCIDP    2632    A
COMCIDP  *                                                                          COMCIDP    2633    A
COMCIDP  *           2. DETERMINE WHETHER OR NOT WE NEED TO XEQ SOMETHING           COMCIDP    2634    A
COMCIDP  *              BEFORE RETURNING CONTROL TO THE CALLER.                     COMCIDP    2635    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *                                                                    COMCIDP    2636    A
COMCIDP  *              THERE ARE 2 FLAVOURS OF THIS --                       COMCIDP    2637    A
COMCIDP  *                                                                    COMCIDP    2638    A
COMCIDP  *              A. IF THIS IS AN INTERACTIVE BREAK (I.E. NOT AN ASSEMBLED  COMCIDP  2639  A
COMCIDP  *                 BREAK), THEN WE MIGHT NEED TO XEQ THE PREVIOUS     COMCIDP    2640    A
COMCIDP  *                 CONTENTS OF THE BREAK ADDR, WHICH WAS SET UP BY *BRK*  COMCIDP  2641  A
COMCIDP  *                 TO BE IN (BC=BRAD).                                COMCIDP    2642    A
COMCIDP  *                                                                    COMCIDP    2643    A
COMCIDP  *              B. IF *STEP* MODE LEFT US IN THE MIDDLE OF A WORD, THEN  COMCIDP  2644  A
COMCIDP  *                 WE NEED TO FINISH UP THE INSTRUCTIONS IN THAT WORD COMCIDP    2645    A
COMCIDP  *                 BECAUSE WE CAN NOT START UP THE CPU IN THE MIDDLE  COMCIDP    2646    A
COMCIDP  *                 OF A WORD.                                         COMCIDP    2647    A
COMCIDP  *                                                                    COMCIDP    2648    A
COMCIDP  *              IT SHOULD BE RELATIVELY CLEAR THAT WE MUST FINISH UP ANY  COMCIDP  2649  A
COMCIDP  *              PARTIAL WORDS SO THAT WE CAN TRANSFER CONTROL TO A FORCED  COMCIDP  2650  A
COMCIDP  *              UPPER INSTRUCTION.  HOWEVER, IT MAY NOT BE QUITE AS CLEAR  COMCIDP  2651  A
COMCIDP  *              WHY WE **MUST** STEP THESE INSTRUCTIONS (AS OPPOSED TO  COMCIDP  2652  A
COMCIDP  *              MERELY NO-OP FILLING A WORD AND XEQING IT).  IDP USED TO  COMCIDP  2653  A
COMCIDP  *              PLAY THE NO-OP FILLING GAME, BUT A NUMBER OF BUGS WERE  COMCIDP  2654  A
COMCIDP  *              DISCOVERED WITH THIS SCHEME...                        COMCIDP    2655    A
COMCIDP  *                                                                    COMCIDP    2656    A
COMCIDP  *              BOTH HAD TO DO WITH THE *RJ* INSTRUCTION.  IT TURNED OUT  COMCIDP  2657  A
COMCIDP  *              THAT UNDER NOT SO UNUSUAL CIRCUMSTANCES, IDP WOULD DO  COMCIDP  2658  A
COMCIDP  *              VERY UGLY THINGS WHEN THIS *FINAL* INSTRUCTION WAS AN  COMCIDP  2659  A
COMCIDP  *              *RJ*.  ERGO...                                        COMCIDP    2660    A
COMCIDP                                                                       COMCIDP    2661    A
COMCIDP                                                                       COMCIDP    2662    A
COMCIDP   ST=END   BSS    0          ** ENTRY **                              COMCIDP    2663    A
COMCIDP            SA1    IDPFLG                                              COMCIDP    2664    A
COMCIDP            LX1    59-IDF.BRFP                                         COMCIDP    2665    A
COMCIDP            MI     X1,ST=END2  IF IN *BRIEF* MODE, NO BYE BYE PROMPT...  COMCIDP  2666  A
COMCIDP                                                                       COMCIDP    2667    A
COMCIDP            PRIDP  (=C= BYE...=),,L.PRB                                COMCIDP    2668    A
COMCIDP                                                                       COMCIDP    2669    A
COMCIDP   ST=END2  SB2    F.IDO                                               COMCIDP    2670    A
COMCIDP            RJ     FOF         FLUSH OUTPUT FILE                       COMCIDP    2671    A
COMCIDP            SB2    =YF.BDO                                             COMCIDP    2672    A
COMCIDP            RJ     FOF         FLUSH OUTPUT FILE                       COMCIDP    2673    A
COMCIDP                                                                       COMCIDP    2674    A
COMCIDP   ST=END3  BSS    0          ** ENTRY FROM SOME PLACES **             COMCIDP    2675    A
COMCIDP            SA1    IDPXLST                                             COMCIDP    2676    A
COMCIDP            SA2    BC=BRAD     (X2) = PREVIOUS CONTENTS OF BREAK ADDR  COMCIDP    2677    A
COMCIDP            BX6    X6-X6                                               COMCIDP    2678    A
COMCIDP            LX7    X1                                                  COMCIDP    2679    A
COMCIDP            SA6    A1+         SET TO *NOT LISTING STEPPED INSTRUCTIONS*  COMCIDP  2680  A
COMCIDP            SA7    ST=ENDA     SAVE ORIGINAL (IDPXLST)                 COMCIDP    2681    A
COMCIDP            NZ     X2,ST=END5  IF WE NEED TO XEQ SOMETHING BEFORE EXITING  COMCIDP  2682  A
COMCIDP                                                                       COMCIDP    2683    A
COMCIDP   ST=END4  SA1    IDPPOS                                              COMCIDP    2684    A
COMCIDP            SX6    X1-60D                                              COMCIDP    2685    A
COMCIDP            ZR     X6,ST=END6  IF WE ARE FORCED UPPER, THEN DONE...    COMCIDP    2686    A
COMCIDP            PL     X6,*+4S15   IF *POS* COUNTER IS BAD...              COMCIDP    2687    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP                                                                                  COMCIDP     2688     A
COMCIDP   ST=END5  RJ     STP         STEP ONE INSTRUCTION                               COMCIDP     2689     A
COMCIDP            EQ     ST=END4                                                        COMCIDP     2690     A
COMCIDP                                                                                  COMCIDP     2691     A
COMCIDP   ST=END6  SA1    ST=ENDA     (X1) = SAVED (IDPXLST)                             COMCIDP     2692     A
COMCIDP            BX6    X1                                                             COMCIDP     2693     A
COMCIDP            SA6    IDPXLST     RESTORE (IDPXLST) TO ORIGINAL VALUE                COMCIDP     2694     A
COMCIDP            EQ     IDP=XIT                                                        COMCIDP     2695     A
COMCIDP                                                                                  COMCIDP     2696     A
COMCIDP                                                                                  COMCIDP     2697     A
COMCIDP   ST=ENDA  BSSZ   1           SAVED (IDPXLST)                                    COMCIDP     2698     A
COMCIDP   ST=FRZ   SPACE  4,10                                                           COMCIDP     2699     A
COMCIDP  *         HERE TO PROCESS *FREEZE*.                                             COMCIDP     2700     A
COMCIDP                                                                                  COMCIDP     2701     A
COMCIDP   ST=FRZ   BSS    0                                                              COMCIDP     2702     A
COMCIDP            SA1    FW.IFR+/IFR/F.FRZ-RA.ORG                                       COMCIDP     2703     A
COMCIDP            SA2    =0LFRZ                                                         COMCIDP     2704     A
COMCIDP            MX3    -18                                                            COMCIDP     2705     A
COMCIDP            BX1    -X3*X1      CLEAR OLD LFN                                      COMCIDP     2706     A
COMCIDP            IX6    X1+X2       MERGE NEW LFN                                      COMCIDP     2707     A
COMCIDP            SA6    A1                                                             COMCIDP     2708     A
COMCIDP            RJ     FRZ         FREEZE INTERACTIVE HOST                            COMCIDP     2709     A
COMCIDP            RJ     UFO         USER FREEZE OWNCODE                                COMCIDP     2710     A
COMCIDP            EQ     IDP=MN                                                         COMCIDP     2711     A
COMCIDP   ST=JP    SPACE  4,10                                                           COMCIDP     2712     A
COMCIDP  *         HERE TO PROCESS *JUMP ADDR*.                                          COMCIDP     2713     A
COMCIDP                                                                                  COMCIDP     2714     A
COMCIDP   ST=JP    BSS    0                                                              COMCIDP     2715     A
COMCIDP            ZR     X5,IDP=MN   IF EOS ENCOUNTERED                                 COMCIDP     2716     A
COMCIDP            RJ     PAS         PARSE SUBEXPRESSION-- ADDR                         COMCIDP     2717     A
COMCIDP            ZR     X3,IDP=MN   IF ADDR IS NULL                                    COMCIDP     2718     A
COMCIDP            SB2    X1                                                             COMCIDP     2719     A
COMCIDP            RJ     CHK         CHECK CM ADDRESS                                   COMCIDP     2720     A
COMCIDP            MI     B2,E.JP     IF ADDR IS BAD                                     COMCIDP     2721     A
COMCIDP            SA1    =46000460004600046000B                                         COMCIDP     2722     A
COMCIDP            SX7    60                                                             COMCIDP     2723     A
COMCIDP            SA7    IDPPOS                                                         COMCIDP     2724     A
COMCIDP            SX6    B2                                                             COMCIDP     2725     A
COMCIDP            BX7    X1                                                             COMCIDP     2726     A
COMCIDP            SA6    IDPPREG                                                        COMCIDP     2727     A
COMCIDP            SA7    BC=BRAD                                                        COMCIDP     2728     A
COMCIDP            EQ     ST=END                                                         COMCIDP     2729     A
COMCIDP   ST=OPT   SPACE  4,10                                                           COMCIDP     2730     A
COMCIDP  **        HERE TO PROCESS *OPTION P1,...PN*.                                    COMCIDP     2731     A
COMCIDP                                                                                  COMCIDP     2732     A
COMCIDP                                                                                  COMCIDP     2733     A
COMCIDP   ST=OPT   BSS    0           ** ENTRY **                                        COMCIDP     2734     A
COMCIDP            ZR     X5,IDP=MN   IF EOS ENCOUNTERED                                 COMCIDP     2735     A
COMCIDP            SB6    FW.KOPT     (B6) = FWA OF *OPTION* SUBKEYWORD TABLE            COMCIDP     2736     A
COMCIDP            RJ     POL         PROCESS OPTIONS LIST ITEM                          COMCIDP     2737     A
COMCIDP            SA1    B6+B7       (X1) = *FOUND* OPTIONS SUBKEYWORD ENTRY            COMCIDP     2738     A
COMCIDP            ZR     X3,ST=OPT   IF NULL PARAMETER, IGNORE...                       COMCIDP     2739     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            SB5    X1          (B5) = OPTIONS PROCESSOR ADDR                    COMCIDP    2740    A
COMCIDP            MX0    1                                                            COMCIDP    2741    A
COMCIDP            SA2    IDPFLG                                                       COMCIDP    2742    A
COMCIDP            BX6    -X6*X0      (X6) = +0 IF -OPTION, OR 1S59 IF +OPTION         COMCIDP    2743    A
COMCIDP            JP     B5                                                           COMCIDP    2744    A
COMCIDP                                                                                COMCIDP    2745    A
COMCIDP                                                                                COMCIDP    2746    A
COMCIDP  *         HERE IF *OPTION (-)MEM*.                                            COMCIDP    2747    A
COMCIDP                                                                                COMCIDP    2748    A
COMCIDP   ST=OPT2  LX0    IDF.MEMP-59                                                  COMCIDP    2749    A
COMCIDP            BX2    -X0*X2      CLEAR OLD *MEM*                                  COMCIDP    2750    A
COMCIDP            LX6    IDF.MEMP-59                                                  COMCIDP    2751    A
COMCIDP            BX6    X2+X6       MERGE NEW *MEM*                                  COMCIDP    2752    A
COMCIDP            SA6    A2                                                           COMCIDP    2753    A
COMCIDP            EQ     ST=OPT                                                       COMCIDP    2754    A
COMCIDP                                                                                COMCIDP    2755    A
COMCIDP                                                                                COMCIDP    2756    A
COMCIDP  *         HERE IF *OPTION (-)XJR*.                                            COMCIDP    2757    A
COMCIDP                                                                                COMCIDP    2758    A
COMCIDP   ST=OPT3  LX0    IDF.XJRP-59                                                  COMCIDP    2759    A
COMCIDP            BX2    -X0*X2      CLEAR OLD *XJR*                                  COMCIDP    2760    A
COMCIDP            LX6    IDF.XJRP-59                                                  COMCIDP    2761    A
COMCIDP            BX6    X2+X6       MERGE NEW *XJR*                                  COMCIDP    2762    A
COMCIDP            SA6    A2                                                           COMCIDP    2763    A
COMCIDP            EQ     ST=OPT                                                       COMCIDP    2764    A
COMCIDP   ST=OUT   SPACE  4,10                                                          COMCIDP    2765    A
COMCIDP  **        HERE TO PROCESS *OUTPUT I,B*.                                       COMCIDP    2766    A
COMCIDP                                                                                COMCIDP    2767    A
COMCIDP   ST=OUT   ZR     X5,IDP=MN   IF EOS ENCOUNTERED                               COMCIDP    2768    A
COMCIDP            SB6    IDPKOUT     (B6) = FWA OF *OUTPUT* OPTIONS KEYWORD TBL       COMCIDP    2769    A
COMCIDP            RJ     POL         PROCESS OPTIONS LIST ITEM                        COMCIDP    2770    A
COMCIDP            SA1    B6+B7                                                        COMCIDP    2771    A
COMCIDP            ZR     X3,ST=OUT   IF NULL PARAMETER, IGNORE...                     COMCIDP    2772    A
COMCIDP            SB5    X1          (B5) = OPTIONS PROCESSOR ADDR                    COMCIDP    2773    A
COMCIDP            MX0    1                                                            COMCIDP    2774    A
COMCIDP            SA2    IDPFLG                                                       COMCIDP    2775    A
COMCIDP            BX6    -X6*X0      (X6) = +0 IF -OPTION, OR 1S59 IF +OPTION         COMCIDP    2776    A
COMCIDP            JP     B5+                                                          COMCIDP    2777    A
COMCIDP                                                                                COMCIDP    2778    A
COMCIDP  *         HERE IF *OUTPUT I* OR *OUTPUT -I*.                                  COMCIDP    2779    A
COMCIDP                                                                                COMCIDP    2780    A
COMCIDP   ST=OUT2  LX0    IDF.IDOP-59                                                  COMCIDP    2781    A
COMCIDP            BX2    -X0*X2      CLEAR EXISTING FIELD                             COMCIDP    2782    A
COMCIDP            LX6    IDF.IDOP-59                                                  COMCIDP    2783    A
COMCIDP            BX6    X2+X6                                                        COMCIDP    2784    A
COMCIDP            SA6    A2                                                           COMCIDP    2785    A
COMCIDP            EQ     ST=OUT                                                       COMCIDP    2786    A
COMCIDP                                                                                COMCIDP    2787    A
COMCIDP  *         HERE IF *OUTPUT B* OR *OUTPUT -B*.                                  COMCIDP    2788    A
COMCIDP                                                                                COMCIDP    2789    A
COMCIDP   ST=OUT3  LX0    IDF.BDOP-59                                                  COMCIDP    2790    A
COMCIDP            BX2    -X0*X2      CLEAR EXISTING FIELD                             COMCIDP    2791    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            LX6    IDF.BDOP-59                                          COMCIDP    2792    A
COMCIDP            BX6    X2+X6                                                COMCIDP    2793    A
COMCIDP            SA6    A2                                                   COMCIDP    2794    A
COMCIDP            EQ     ST=OUT                                               COMCIDP    2795    A
COMCIDP                                                                        COMCIDP    2796    A
COMCIDP  *         HERE IF *OUTPUT BE* OR *OUTPUT -BE*.                         COMCIDP    2797    A
COMCIDP                                                                        COMCIDP    2798    A
COMCIDP   ST=OUT4  LX0    IDF.BKOP-59                                          COMCIDP    2799    A
COMCIDP            BX2    -X0*X2       CLEAR EXISTING FIELD                    COMCIDP    2800    A
COMCIDP            LX6    IDF.BKOP-59                                          COMCIDP    2801    A
COMCIDP            BX6    X2+X6        MERGE NEW BKO                           COMCIDP    2802    A
COMCIDP            SA6    A2                                                   COMCIDP    2803    A
COMCIDP            EQ     ST=OUT                                               COMCIDP    2804    A
COMCIDP                                                                        COMCIDP    2805    A
COMCIDP  *         HERE IF *OUTPUT IE* OR *OUTPUT -IE*.                         COMCIDP    2806    A
COMCIDP                                                                        COMCIDP    2807    A
COMCIDP   ST=OUT5  LX0    IDF.IKOP-59                                          COMCIDP    2808    A
COMCIDP            BX2    -X0*X2       CLEAR EXISTING FIELD                    COMCIDP    2809    A
COMCIDP            LX6    IDF.IKOP-59                                          COMCIDP    2810    A
COMCIDP            BX6    X2+X6        MERGE NEW IKO                           COMCIDP    2811    A
COMCIDP            SA6    A2                                                   COMCIDP    2812    A
COMCIDP            EQ     ST=OUT                                               COMCIDP    2813    A
COMCIDP   ST=REG   SPACE  4,10                                                 COMCIDP    2814    A
COMCIDP  *         HERE TO PROCESS *REG R1,R2,...,RN*.                          COMCIDP    2815    A
COMCIDP                                                                        COMCIDP    2816    A
COMCIDP   ST=REG   BSS    0                                                    COMCIDP    2817    A
COMCIDP            SA0    A5           INITIALIZE (A0) = ADDR OF CURRENT TOKEN COMCIDP    2818    A
COMCIDP            NZ     X5,ST=REG2   IF *REG* NOT FOLLOWED BY EOS            COMCIDP    2819    A
COMCIDP            RJ     DAR          DUMP ALL REGISTERS                      COMCIDP    2820    A
COMCIDP            EQ     IDP=MN                                               COMCIDP    2821    A
COMCIDP                                                                        COMCIDP    2822    A
COMCIDP  *         HERE FOR SELECTED REGISTER DUMP.                             COMCIDP    2823    A
COMCIDP                                                                        COMCIDP    2824    A
COMCIDP   ST=REG2  SA0    A0+B1                                                COMCIDP    2825    A
COMCIDP            SA5    A0                                                   COMCIDP    2826    A
COMCIDP            ZR     X5,IDP=MN    IF AN EOS ENCOUNTERED                   COMCIDP    2827    A
COMCIDP                                                                        COMCIDP    2828    A
COMCIDP            SX6    X5-O.COM                                             COMCIDP    2829    A
COMCIDP            SX7    X5-O.CONS                                            COMCIDP    2830    A
COMCIDP            SA4    IDP0TR                                               COMCIDP    2831    A
COMCIDP            ZR     X6,ST=REG2   IF COMMA TOKEN                          COMCIDP    2832    A
COMCIDP            ZR     X7,ST=REG3   IF CONSTANT TOKEN                       COMCIDP    2833    A
COMCIDP            SX6    X5-O.REG                                             COMCIDP    2834    A
COMCIDP            NZ     X6,E.REG     IF NOT *REG* TOKEN, ERROR...            COMCIDP    2835    A
COMCIDP                                                                        COMCIDP    2836    A
COMCIDP  *         HERE IF A *REG* TOKEN ENCOUNTERED.                           COMCIDP    2837    A
COMCIDP                                                                        COMCIDP    2838    A
COMCIDP            LX5    0-TB.0TRP                                            COMCIDP    2839    A
COMCIDP            SB4    X5           (B4) = 0TR                              COMCIDP    2840    A
COMCIDP            EQ     ST=REG4                                              COMCIDP    2841    A
COMCIDP                                                                        COMCIDP    2842    A
COMCIDP  *         HERE IF A *CONS* TOKEN ENCOUNTERED.                          COMCIDP    2843    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP                                                                                COMCIDP    2844    A
COMCIDP                                                                                COMCIDP    2845    A
COMCIDP   ST=REG3  LX5    CHAR                                                         COMCIDP    2846    A
COMCIDP            MX0    -CHAR                                                        COMCIDP    2847    A
COMCIDP            BX6    -X0*X5                                                       COMCIDP    2848    A
COMCIDP            MX1    1R7-1R0+1                                                    COMCIDP    2849    A
COMCIDP            LX1    -1R0        (X1) = (0-7) SHIFT MASK                          COMCIDP    2850    A
COMCIDP            SB3    X6          (B3) = REGISTER NR (0-7)                         COMCIDP    2851    A
COMCIDP            LX7    X1,B3                                                        COMCIDP    2852    A
COMCIDP            LX5    5*CHAR                                                       COMCIDP    2853    A
COMCIDP            PL     X7,E.REG    IF NOT (0-7), ERROR...                           COMCIDP    2854    A
COMCIDP            MX0    -5*CHAR                                                      COMCIDP    2855    A
COMCIDP            BX7    -X0*X5                                                       COMCIDP    2856    A
COMCIDP            NZ     X7,E.REG    IF REST OF CHARS NON-ZERO, NOT A REGISTER        COMCIDP    2857    A
COMCIDP            SB6    B3-1R0      (B6) = REGISTER NR IN BINARY                     COMCIDP    2858    A
COMCIDP            MX0    60-3                                                         COMCIDP    2859    A
COMCIDP            BX7    X0*X4       STRIP OLD REGISTER NR                            COMCIDP    2860    A
COMCIDP            SB4    X7+B6       (B4) = NEW 0TR                                   COMCIDP    2861    A
COMCIDP                                                                                COMCIDP    2862    A
COMCIDP   ST=REG4  SX6    B4                                                           COMCIDP    2863    A
COMCIDP            SA6    IDP0TR                                                       COMCIDP    2864    A
COMCIDP            SB7    020B                                                         COMCIDP    2865    A
COMCIDP            GE     B4,B7,ST=REG5     IF DUMPING X REGISTER                      COMCIDP    2866    A
COMCIDP            RJ     DAB         DUMP -A- OR -B- REGISTER                         COMCIDP    2867    A
COMCIDP            EQ     ST=REG2                                                      COMCIDP    2868    A
COMCIDP                                                                                COMCIDP    2869    A
COMCIDP   ST=REG5  RJ     DUX         DUMP -X- REGISTER                                COMCIDP    2870    A
COMCIDP            EQ     ST=REG2                                                      COMCIDP    2871    A
COMCIDP   ST=REL   SPACE  4,10                                                         COMCIDP    2872    A
COMCIDP  **        HERE TO PROCESS *REL*.                                              COMCIDP    2873    A
COMCIDP                                                                                COMCIDP    2874    A
COMCIDP                                                                                COMCIDP    2875    A
COMCIDP   ST=REL   BSS    0           ** ENTRY **                                      COMCIDP    2876    A
COMCIDP            SA1    IDPFLG                                                       COMCIDP    2877    A
COMCIDP            MX2    1                                                            COMCIDP    2878    A
COMCIDP            LX2    IDF.ADRP-59                                                  COMCIDP    2879    A
COMCIDP            BX6    X1+X2       SET *ADR*                                        COMCIDP    2880    A
COMCIDP            SA6    A1                                                           COMCIDP    2881    A
COMCIDP            EQ     IDP=MN                                                       COMCIDP    2882    A
COMCIDP   ST=SET   SPACE  4,10                                                         COMCIDP    2883    A
COMCIDP  *         HERE TO PROCESS *SET NAME,ADDR*.                                    COMCIDP    2884    A
COMCIDP                                                                                COMCIDP    2885    A
COMCIDP   ST=SET   ZR     X5,ST=SET3  IF EOS ENCOUNTERED                               COMCIDP    2886    A
COMCIDP            SA5    A5+B1       (X5) = 42/0LNAME,18/0.VAR                        COMCIDP    2887    A
COMCIDP            SX6    X5-0.VAR                                                     COMCIDP    2888    A
COMCIDP            NZ     X6,E.SET    IF NOT A NAME TOKEN                              COMCIDP    2889    A
COMCIDP            MX0    7*CHAR                                                       COMCIDP    2890    A
COMCIDP            BX7    X0*X5                                                        COMCIDP    2891    A
COMCIDP            SA7    IDPNAM      SAVE 0LNAME                                      COMCIDP    2892    A
COMCIDP            SA5    A5+B1                                                        COMCIDP    2893    A
COMCIDP            RJ     PAS         PARSE SUBEXPRESSION -- ADDR                      COMCIDP    2894    A
COMCIDP            MX0    7*CHAR                                                       COMCIDP    2895    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            SB6    IDPSET      (B6) = FWA OF *SET* TABLE                    COMCIDP    2896    A
COMCIDP            BX4    -X0*X1      (X4) = 42/0,18/VALUE OF *NAME*               COMCIDP    2897    A
COMCIDP            SA1    IDPNAM                                                   COMCIDP    2898    A
COMCIDP            RJ     SKT         SEARCH *SET* TABLE FOR NAME                  COMCIDP    2899    A
COMCIDP            IX6    X1+X4       (X6) = 42/0LNAME, 18/VALUE OF NAME           COMCIDP    2900    A
COMCIDP            ZR     X2,ST=SET2  IF NO FIND                                   COMCIDP    2901    A
COMCIDP            SA6    A2                                                       COMCIDP    2902    A
COMCIDP            EQ     IDP=MN                                                   COMCIDP    2903    A
COMCIDP                                                                            COMCIDP    2904    A
COMCIDP   ST=SET2  RJ     ADZ         ADD A WORD TO IDP TABLE                      COMCIDP    2905    A
COMCIDP            EQ     IDP=MN                                                   COMCIDP    2906    A
COMCIDP                                                                            COMCIDP    2907    A
COMCIDP   *        HERE TO LIST *SET* TABLE.                                       COMCIDP    2908    A
COMCIDP                                                                            COMCIDP    2909    A
COMCIDP   ST=SET3  SA5    IDPSET                                                   COMCIDP    2910    A
COMCIDP            RJ     LST         LIST SET TABLE                               COMCIDP    2911    A
COMCIDP            EQ     IDP=MN                                                   COMCIDP    2912    A
COMCIDP   ST=SNP   SPACE  4,10                                                     COMCIDP    2913    A
COMCIDP   *        HERE TO PROCESS *SNAP FWA,LWA,LEN*.                             COMCIDP    2914    A
COMCIDP                                                                            COMCIDP    2915    A
COMCIDP   ST=SNP   BSS    0                                                        COMCIDP    2916    A
COMCIDP            RJ     PAT         PARSE *FWA,LWA,LEN* TRIPLE                   COMCIDP    2917    A
COMCIDP            MI     B2,E.FLL    IF FWA, LWA, OR LEN IS BAD                   COMCIDP    2918    A
COMCIDP            RJ     DCM         DUMP CENTRAL MEMORY                          COMCIDP    2919    A
COMCIDP            PL     B2,IDP=MN   IF FWA,LWA,AND LEN OK                        COMCIDP    2920    A
COMCIDP            EQ     SE.FLL                                                   COMCIDP    2921    A
COMCIDP   ST=STA   SPACE  4,10                                                     COMCIDP    2922    A
COMCIDP   **       HERE TO PROCESS *STATUS*.                                       COMCIDP    2923    A
COMCIDP                                                                            COMCIDP    2924    A
COMCIDP   ST=STA   BSS    0           ** ENTRY **                                  COMCIDP    2925    A
COMCIDP            SA3    IDPFLG                                                   COMCIDP    2926    A
COMCIDP            SA2    ST=STAA     (X2) = 10L_ IN IDP                           COMCIDP    2927    A
COMCIDP            LX3    59-IDF.BRFP                                              COMCIDP    2928    A
COMCIDP            BX6    X2                                                       COMCIDP    2929    A
COMCIDP            SB7    SNAPLNE     (B7) = FWA OF OUTPUT LINE IMAGE AREA         COMCIDP    2930    A
COMCIDP            SA1    A2+B1       (X1) = 10L_, *BRIEF*                         COMCIDP    2931    A
COMCIDP            SA6    B7                                                       COMCIDP    2932    A
COMCIDP            PL     X3,ST=STA2  IF NOT IN *BRIEF* MODE                       COMCIDP    2933    A
COMCIDP            RJ     =XMCS       MERGE CODED STRINGS                          COMCIDP    2934    A
COMCIDP                                                                            COMCIDP    2935    A
COMCIDP   ST=STA2  BX6    X6-X6                                                    COMCIDP    2936    A
COMCIDP            SA6    B7+B1       MARK EOL                                     COMCIDP    2937    A
COMCIDP            PRIDP  SNAPLNE                                                  COMCIDP    2938    A
COMCIDP            EQ     IDP=MN                                                   COMCIDP    2939    A
COMCIDP                                                                            COMCIDP    2940    A
COMCIDP                                                                            COMCIDP    2941    A
COMCIDP   ST=STAA  DATA   L. IN IDP.                                               COMCIDP    2942    A
COMCIDP            DATA   L., *BRIEF*.                                             COMCIDP    2943    A
COMCIDP   ST=STO   SPACE  4,10                                                     COMCIDP    2944    A
COMCIDP   *        HERE TO PROCESS *STORE ADDR,C1,C2,C3,C4*.                       COMCIDP    2945    A
COMCIDP                                                                            COMCIDP    2946    A
COMCIDP   ST=STO   BSS    0                                                        COMCIDP    2947    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            ZR     X5,E.STO    IF EOS ENCOUNTERED                          COMCIDP    2948    A
COMCIDP            RJ     PAS         PARSE SUBEXPRESSION -- ADDR TO STORE        COMCIDP    2949    A
COMCIDP            ZR     X3,E.STO    IF 1ST EXPRESSION *ADDR* IS NULL            COMCIDP    2950    A
COMCIDP            SX6    X1                                                      COMCIDP    2951    A
COMCIDP            SB2    X1                                                      COMCIDP    2952    A
COMCIDP            SA6    IDPSTA      SAVE STORE ADDR                             COMCIDP    2953    A
COMCIDP            RJ     CHK         CHECK CM ADDRESS -- STORE ADDR              COMCIDP    2954    A
COMCIDP            MI     B2,E.STO    IF ADDR IS BAD                              COMCIDP    2955    A
COMCIDP            MX6    0                                                       COMCIDP    2956    A
COMCIDP            SA6    A6+B1       CLEAR (IDPSTC) = NEW VALUE TO STORE         COMCIDP    2957    A
COMCIDP            SA6    A6+B1       CLEAR (IDPSTM) = NULL PARAMETER MASK        COMCIDP    2958    A
COMCIDP            ZR     X5,ST=STO4  IF STORE ADDR FOLLOWED BY EOS               COMCIDP    2959    A
COMCIDP                                                                           COMCIDP    2960    A
COMCIDP   ST=STO2  RJ     PAS         PARSE SUBEXPRESSION -- C-N-                 COMCIDP    2961    A
COMCIDP            SA2    IDPSTC      (X2) = CURRENT VALUE TO BE STORED           COMCIDP    2962    A
COMCIDP            SA4    A2+B1       (X4) = CURRENT NULL PARAMETER MASK          COMCIDP    2963    A
COMCIDP            MX0    -15                                                     COMCIDP    2964    A
COMCIDP            LX2    15                                                      COMCIDP    2965    A
COMCIDP            LX4    15                                                      COMCIDP    2966    A
COMCIDP            BX6    X2+X1                                                   COMCIDP    2967    A
COMCIDP            LX7    X4                                                      COMCIDP    2968    A
COMCIDP            SA6    A2                                                      COMCIDP    2969    A
COMCIDP            NZ     X3,ST=STO3  IF C-N- NOT NULL                            COMCIDP    2970    A
COMCIDP            BX7    -X0+X4                                                  COMCIDP    2971    A
COMCIDP                                                                           COMCIDP    2972    A
COMCIDP   ST=STO3  SA7    A4                                                      COMCIDP    2973    A
COMCIDP            NZ     X5,ST=STO2  IF NOT FINISHED ASSEMBLING STORE VALUE      COMCIDP    2974    A
COMCIDP                                                                           COMCIDP    2975    A
COMCIDP   *        HERE IF FINISHED ASSEMBLING STORE VALUE.                       COMCIDP    2976    A
COMCIDP                                                                           COMCIDP    2977    A
COMCIDP   ST=STO4  SA1    IDPSTA      (X1) = STORE ADDR                           COMCIDP    2978    A
COMCIDP            SA2    A1+B1       (X2) = NEW ASSEMBLED STORE VALUE            COMCIDP    2979    A
COMCIDP            SA3    A2+B1       (X3) = NULL PARAMETER MASK                  COMCIDP    2980    A
COMCIDP            SA1    X1          (X1) = OLD CONTENTS OF STORE ADDR           COMCIDP    2981    A
COMCIDP            BX6    X3*X1                                                   COMCIDP    2982    A
COMCIDP            BX7    X2+X6       (X7) = NEW CONTENTS OF STORE ADDR           COMCIDP    2983    A
COMCIDP            SA7    A1          NEW REPLACES OLD                            COMCIDP    2984    A
COMCIDP            SA7    A2                                                      COMCIDP    2985    A
COMCIDP            RJ     =XWOD       CONVERT A WORD OF BINARY TO OCTAL DPC--OLD  COMCIDP    2986    A
COMCIDP            SA6    IDPMSG4+1                                               COMCIDP    2987    A
COMCIDP            SA7    A6+B1                                                   COMCIDP    2988    A
COMCIDP            SA1    IDPSTC      (X1) = NEW CONTENTS OF STORE ADDR           COMCIDP    2989    A
COMCIDP            RJ     =XWOD       CONVERT A WORD OF BINARY TO OCTAL DPC--NEW  COMCIDP    2990    A
COMCIDP            SA6    IDPMSG5+1                                               COMCIDP    2991    A
COMCIDP            SA7    A6+B1                                                   COMCIDP    2992    A
COMCIDP            PRIDP  IDPMSG4,4                                               COMCIDP    2993    A
COMCIDP            PRIDP  IDPMSG5,4                                               COMCIDP    2994    A
COMCIDP            EQ     IDP=MN                                                  COMCIDP    2995    A
COMCIDP   ST=STP   SPACE  4,10                                                    COMCIDP    2996    A
COMCIDP   *        HERE TO PROCESS *STEP L,RJ*,                                   COMCIDP    2997    A
COMCIDP   *                   OR *STPL ADDR1,L,RJ* FORM,                          COMCIDP    2998    A
COMCIDP   *                   OR *STEQ ADDR1,ADDR2,L,RJ* FORM.                    COMCIDP    2999    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP                                                                                  COMCIDP     3000     A
COMCIDP   ST=STP   BSS    0           ** ENTRY **                                        COMCIDP     3001     A
COMCIDP            MX2    -KW.XATL                                                       COMCIDP     3002     A
COMCIDP            LX1    0-KW.XATP                                                      COMCIDP     3003     A
COMCIDP            BX6    -X2*X1      (X6) = STEP TYPE CODE (XTC)                        COMCIDP     3004     A
COMCIDP            SA6    IDPXTC                                                         COMCIDP     3005     A
COMCIDP            ZR     X6,ST=STP5  IF *STEP L,RJ* FORM                                COMCIDP     3006     A
COMCIDP            ZR     X5,ST=STP1  IF EOS ENCOUNTERED                                 COMCIDP     3007     A
COMCIDP            RJ     PAS         PARSE SUBEXPRESSION-- ADDR1                        COMCIDP     3008     A
COMCIDP            NZ     X3,ST=STP2  IF ADDR1 NOT NULL                                  COMCIDP     3009     A
COMCIDP                                                                                  COMCIDP     3010     A
COMCIDP   ST=STP1  SX1    0           (X1) = DEFAULT ADDR1                               COMCIDP     3011     A
COMCIDP                                                                                  COMCIDP     3012     A
COMCIDP   ST=STP2  SB2    X1                                                             COMCIDP     3013     A
COMCIDP            RJ     CHK         CHECK CM ADDRESS                                   COMCIDP     3014     A
COMCIDP            MI     B2,E.BRA1   IF ADDR1 IS BAD                                    COMCIDP     3015     A
COMCIDP            SX6    B2                                                             COMCIDP     3016     A
COMCIDP            MX3    0                                                              COMCIDP     3017     A
COMCIDP            SA6    IDPXA1                                                         COMCIDP     3018     A
COMCIDP            SA2    IDPXTC                                                         COMCIDP     3019     A
COMCIDP            SX7    X2-XTC.EQ                                                      COMCIDP     3020     A
COMCIDP            MI     X7,ST=STP3  IF THIS IS *STPL ADDR1,L,RJ* FORM                  COMCIDP     3021     A
COMCIDP                                                                                  COMCIDP     3022     A
COMCIDP   *        ASSEMBLE ADDR2.                                                       COMCIDP     3023     A
COMCIDP                                                                                  COMCIDP     3024     A
COMCIDP            ZR     X5,ST=STP3  IF ADDR1 FOLLOWED BY EOS                           COMCIDP     3025     A
COMCIDP            RJ     PAS         PARSE SUBEXPRESSION-- ADDR2                        COMCIDP     3026     A
COMCIDP                                                                                  COMCIDP     3027     A
COMCIDP   ST=STP3  NZ     X3,ST=STP4  IF ADDR2 NOT NULL                                  COMCIDP     3028     A
COMCIDP            SA1    IDPXA1      (X1) = DEFAULT ADDR2                               COMCIDP     3029     A
COMCIDP                                                                                  COMCIDP     3030     A
COMCIDP   ST=STP4  SB2    X1                                                             COMCIDP     3031     A
COMCIDP            SX6    B2                                                             COMCIDP     3032     A
COMCIDP            RJ     CHK         CHECK CM ADDRESS                                   COMCIDP     3033     A
COMCIDP            MI     B2,E.BRA2   IF ADDR2 IS BAD                                    COMCIDP     3034     A
COMCIDP            SA6    IDPXA2                                                         COMCIDP     3035     A
COMCIDP                                                                                  COMCIDP     3036     A
COMCIDP   *        HERE TO PROCESS *L,RJ* OPTIONS LIST.                                  COMCIDP     3037     A
COMCIDP                                                                                  COMCIDP     3038     A
COMCIDP   ST=STP5  ZR     X5,ST=STP6  IF EOS ENCOUNTERED                                 COMCIDP     3039     A
COMCIDP            SB6    IDPKSTO     (B6) = FWA OF STEP OPTIONS KEYWORD TABLE           COMCIDP     3040     A
COMCIDP            RJ     POL         PROCESS OPTIONS LIST                               COMCIDP     3041     A
COMCIDP            ZR     X3,ST=STP5  IF NULL PARAMETER                                  COMCIDP     3042     A
COMCIDP            SA7    IDPXOP+B7                                                      COMCIDP     3043     A
COMCIDP            EQ     ST=STP5                                                        COMCIDP     3044     A
COMCIDP                                                                                  COMCIDP     3045     A
COMCIDP   *        HERE TO SET UP FOR ACTUAL STEP.                                       COMCIDP     3046     A
COMCIDP                                                                                  COMCIDP     3047     A
COMCIDP   ST=STP6  SA1    IDPXTC                                                         COMCIDP     3048     A
COMCIDP            SA2    IDPXA2                                                         COMCIDP     3049     A
COMCIDP            MX6    0                                                              COMCIDP     3050     A
COMCIDP            SX3    X1-XTC.NR                                                      COMCIDP     3051     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            BX7    X2                                                  COMCIDP    3052    A
COMCIDP            SA6    IDPXRJ      SET TO *AT LEVEL 0 NOW*                 COMCIDP    3053    A
COMCIDP            NZ     X3,ST=STP7  IF NOT *STNR NR,L,RJ*                   COMCIDP    3054    A
COMCIDP            SA7    IDPXA1      (RE)SET STEP COUNT                      COMCIDP    3055    A
COMCIDP                                                                       COMCIDP    3056    A
COMCIDP   ST=STP7  SA1    IDPXOL      (X1) = /L/ OPTION FLAG                  COMCIDP    3057    A
COMCIDP            SA2    A1+B1       (X2) = /RJ/ OPTION FLAG                 COMCIDP    3058    A
COMCIDP            SX6    B1          SET TO *LIST ON*                        COMCIDP    3059    A
COMCIDP            SA3    IDPXRJ                                              COMCIDP    3060    A
COMCIDP            NZ     X2,ST=STP7A IF /RJ/ TRACING ON                      COMCIDP    3061    A
COMCIDP            ZR     X3,ST=STP7A IF AT LEVEL 0 (I.E. NO ACTIVE *RJ*)     COMCIDP    3062    A
COMCIDP            SX6    B0          SET TO *LIST OFF*                       COMCIDP    3063    A
COMCIDP                                                                       COMCIDP    3064    A
COMCIDP   ST=STP7A BX7    X6*X1                                               COMCIDP    3065    A
COMCIDP            SA7    IDPXLST                                             COMCIDP    3066    A
COMCIDP                                                                       COMCIDP    3067    A
COMCIDP   *        HERE TO CHECK STEP CONDITIONS.                             COMCIDP    3068    A
COMCIDP                                                                       COMCIDP    3069    A
COMCIDP   ST=STP8  SA3    IDPXTC      (X3) = STEP TYPE CODE                   COMCIDP    3070    A
COMCIDP            SA1    IDPXA1      (X1) = A1                               COMCIDP    3071    A
COMCIDP            SA2    A1+B1       (X2) = A2                               COMCIDP    3072    A
COMCIDP            SB5    X3                                                  COMCIDP    3073    A
COMCIDP            SX6    X3-XTC.NR                                           COMCIDP    3074    A
COMCIDP            SX0    X3-XTC.AR                                           COMCIDP    3075    A
COMCIDP            SX7    X3-XTC.RNG                                          COMCIDP    3076    A
COMCIDP            ZR     X3,ST=STP9  IF *STEP L,RJ*                          COMCIDP    3077    A
COMCIDP            ZR     X6,ST=STP10 IF *STNR NR,L,RJ*                       COMCIDP    3078    A
COMCIDP            ZR     X0,ST=STP9  IF *STAR ADDR1,ADDR2,L,RJ*              COMCIDP    3079    A
COMCIDP            ZR     X7,ST=STP12 IF *STRANGE ADDR1,ADDR2,L,RJ*           COMCIDP    3080    A
COMCIDP                                                                       COMCIDP    3081    A
COMCIDP            SA1    X1          (X1) = C1                               COMCIDP    3082    A
COMCIDP            SA2    X2          (X2) = C2                               COMCIDP    3083    A
COMCIDP            RJ     CBC         CHECK STEP CONDITION                    COMCIDP    3084    A
COMCIDP            ZR     B6,IDP=MN   IF CONDITION NOT SATISFIED              COMCIDP    3085    A
COMCIDP                                                                       COMCIDP    3086    A
COMCIDP   ST=STP9  RJ     STP         STEP ONE INSTRUCTION                    COMCIDP    3087    A
COMCIDP            SA1    IDPXTC                                              COMCIDP    3088    A
COMCIDP            SA2    IDPXRJ                                              COMCIDP    3089    A
COMCIDP            SA3    IDPXORJ                                             COMCIDP    3090    A
COMCIDP            SX6    X1-XTC.AR                                           COMCIDP    3091    A
COMCIDP            ZR     X6,ST=STP9A IF *STAR ADDR1,ADDR2,L,RJ*              COMCIDP    3092    A
COMCIDP            NZ     X1,ST=STP7  IF NOT *STEP L,RJ*                      COMCIDP    3093    A
COMCIDP                                                                       COMCIDP    3094    A
COMCIDP   *        HERE IF *STEP L,RJ*.                                       COMCIDP    3095    A
COMCIDP                                                                       COMCIDP    3096    A
COMCIDP            NZ     X3,IDP=MN   IF /RJ/ SELECTED                        COMCIDP    3097    A
COMCIDP            ZR     X2,IDP=MN   IF AT LEVEL 0                           COMCIDP    3098    A
COMCIDP            EQ     ST=STP7                                             COMCIDP    3099    A
COMCIDP                                                                       COMCIDP    3100    A
COMCIDP   *        HERE TO PROCESS *STAR ADDR1,ADDR2,L,RJ*.                   COMCIDP    3101    A
COMCIDP                                                                       COMCIDP    3102    A
COMCIDP   ST=STP9A SA1    IDPXAR                                              COMCIDP    3103    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            SA2    IDPXA1                                          COMCIDP    3104    A
COMCIDP            SA3    IDPXA2                                          COMCIDP    3105    A
COMCIDP            MI     X1,ST=STP7  IF NO ADDR REF                      COMCIDP    3106    A
COMCIDP            IX6    X1-X2                                           COMCIDP    3107    A
COMCIDP            IX7    X3-X1                                           COMCIDP    3108    A
COMCIDP            MI     X6,ST=STP7  IF ADDR BELOW LOWER LIMIT           COMCIDP    3109    A
COMCIDP            MI     X7,ST=STP7  IF ADDR ABOVE UPPER LIMIT           COMCIDP    3110    A
COMCIDP            EQ     IDP=MN                                          COMCIDP    3111    A
COMCIDP                                                                   COMCIDP    3112    A
COMCIDP  *         HERE TO PROCESS *STNR NR,L,RJ*.                         COMCIDP    3113    A
COMCIDP                                                                   COMCIDP    3114    A
COMCIDP   ST=STP10 SA2    IDPXORJ                                         COMCIDP    3115    A
COMCIDP            SA3    IDPXRJ                                          COMCIDP    3116    A
COMCIDP            SX6    X1-1                                            COMCIDP    3117    A
COMCIDP            NZ     X2,ST=STP11 IF /RJ/ SELECTED                    COMCIDP    3118    A
COMCIDP            NZ     X3,ST=STP9  IF NOT AT LEVEL 0                   COMCIDP    3119    A
COMCIDP                                                                   COMCIDP    3120    A
COMCIDP   ST=STP11 MI     X6,IDP=MN   IF NO MORE INSTRUCTIONS TO STEP     COMCIDP    3121    A
COMCIDP            SA6    A1                                              COMCIDP    3122    A
COMCIDP            EQ     ST=STP9                                         COMCIDP    3123    A
COMCIDP                                                                   COMCIDP    3124    A
COMCIDP  *         HERE TO PROCESS *STRANGE ADDR1,ADDR2,L,RJ*.             COMCIDP    3125    A
COMCIDP                                                                   COMCIDP    3126    A
COMCIDP   ST=STP12 SA4    IDPPREG                                         COMCIDP    3127    A
COMCIDP            IX6    X4-X1                                           COMCIDP    3128    A
COMCIDP            IX7    X2-X4                                           COMCIDP    3129    A
COMCIDP            MI     X6,IDP=MN   IF P BELOW LOWER RANGE              COMCIDP    3130    A
COMCIDP            MI     X7,IDP=MN   IF P ABOVE UPPER RANGE              COMCIDP    3131    A
COMCIDP            EQ     ST=STP9                                         COMCIDP    3132    A
COMCIDP   ST=TIM   SPACE  4,10                                             COMCIDP    3133    A
COMCIDP  **        HERE TO PROCESS *TIME*.                                 COMCIDP    3134    A
COMCIDP                                                                   COMCIDP    3135    A
COMCIDP   ST=TIM   BSS    0           ** ENTRY **                         COMCIDP    3136    A
COMCIDP            CLOCK  SNAPLNE                                         COMCIDP    3137    A
COMCIDP            BX6    X6-X6                                           COMCIDP    3138    A
COMCIDP            SA6    SNAPLNE+1   MARK EOL                            COMCIDP    3139    A
COMCIDP            PRIDP  SNAPLNE,2                                       COMCIDP    3140    A
COMCIDP            EQ     IDP=MN                                          COMCIDP    3141    A
COMCIDP   ST=UBK   SPACE  4,10                                             COMCIDP    3142    A
COMCIDP  *         HERE TO PROCESS *UNBREAK ADDR1,...,ADDRN*.              COMCIDP    3143    A
COMCIDP                                                                   COMCIDP    3144    A
COMCIDP   ST=UBK   BSS    0                                               COMCIDP    3145    A
COMCIDP            ZR     X5,IDP=MN   IF EOS ENCOUNTERED                  COMCIDP    3146    A
COMCIDP            RJ     PAS         PARSE SUBEXPRESSION-- ADDR          COMCIDP    3147    A
COMCIDP            ZR     X3,ST=UBK   IF ADDR IS NULL                     COMCIDP    3148    A
COMCIDP            SB2    X6          (B2) = ADDR OF BREAK TO UNBREAK     COMCIDP    3149    A
COMCIDP            SB3    X6                                              COMCIDP    3150    A
COMCIDP            RJ     UBK         UNBREAK A BREAK                     COMCIDP    3151    A
COMCIDP            EQ     ST=UBK      CONTINUE...                         COMCIDP    3152    A
COMCIDP   ST=UST   SPACE  4,10                                             COMCIDP    3153    A
COMCIDP  *         HERE TO PROCESS *UNSET NAME1,...,NAMEN*.                COMCIDP    3154    A
COMCIDP                                                                   COMCIDP    3155    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP   ST=UST   BSS    0                                                COMCIDP    3156    A
COMCIDP                                                                   COMCIDP    3157    A
COMCIDP   ST=UST2  ZR     X5,IDP=MN   IF EOS ENCOUNTERED                  COMCIDP    3158    A
COMCIDP            SA5    A5+B1                                           COMCIDP    3159    A
COMCIDP            NO                                                     COMCIDP    3160    A
COMCIDP            SX6    X5-O.VAR                                        COMCIDP    3161    A
COMCIDP            NZ     X6,ST=UST2  IF NOT A NAME TOKEN                 COMCIDP    3162    A
COMCIDP            SB6    IDPSET                                          COMCIDP    3163    A
COMCIDP            MX0    7*CHAR                                          COMCIDP    3164    A
COMCIDP            BX1    X5                                              COMCIDP    3165    A
COMCIDP            RJ     SKT         SEARCH FOR NAME                     COMCIDP    3166    A
COMCIDP            ZR     X2,ST=UST2  IF NO FIND                          COMCIDP    3167    A
COMCIDP            SX6    -B1                                             COMCIDP    3168    A
COMCIDP            SA6    A2          SET TO *THIS ENTRY IS AVAILABLE*    COMCIDP    3169    A
COMCIDP            EQ     ST=UST2                                         COMCIDP    3170    A
COMCIDP   ST=WHR   SPACE  4,10                                            COMCIDP    3171    A
COMCIDP  *         HERE TO PROCESS *WHERE ADDR1,...,ADDRN*.               COMCIDP    3172    A
COMCIDP                                                                   COMCIDP    3173    A
COMCIDP   ST=WHR   BSS    0                                               COMCIDP    3174    A
COMCIDP            ZR     X5,ST=WHR2A IF EOS ENCOUNTERED                  COMCIDP    3175    A
COMCIDP                                                                   COMCIDP    3176    A
COMCIDP   ST=WHR2  RJ     PAS         PARSE SUBEXPRESSION                 COMCIDP    3177    A
COMCIDP            NZ     X3,ST=WHR3  IF ADDR NOT NULL                    COMCIDP    3178    A
COMCIDP                                                                   COMCIDP    3179    A
COMCIDP   ST=WHR2A SA1    IDPPREG                                         COMCIDP    3180    A
COMCIDP                                                                   COMCIDP    3181    A
COMCIDP   ST=WHR3  MX0    -18                                             COMCIDP    3182    A
COMCIDP            BX1    -X0*X1                                          COMCIDP    3183    A
COMCIDP            LX0    X1          SAVE (X1) = ADDR                    COMCIDP    3184    A
COMCIDP            RJ     COD         CONVERT BINARY TO OCTAL DPC         COMCIDP    3185    A
COMCIDP            LX6    9*CHAR-6*CHAR                                   COMCIDP    3186    A
COMCIDP            BX1    X0                                              COMCIDP    3187    A
COMCIDP            SA6    SNAPLNE                                         COMCIDP    3188    A
COMCIDP                                                                   COMCIDP    3189    A
COMCIDP            RJ     FRA         FIND RELATIVE ADDRESS               COMCIDP    3190    A
COMCIDP            SA6    A6+B1                                           COMCIDP    3191    A
COMCIDP            SA7    A6+B1                                           COMCIDP    3192    A
COMCIDP            PRIDP  SNAPLNE                                         COMCIDP    3193    A
COMCIDP            NZ     X5,ST=WHR2  IF NOT EOS TOKEN                    COMCIDP    3194    A
COMCIDP            EQ     IDP=MN                                          COMCIDP    3195    A
COMCIDP   ST=XEQ   SPACE  4,10                                            COMCIDP    3196    A
COMCIDP  *         HERE TO PROCESS *XEQ*.                                 COMCIDP    3197    A
COMCIDP                                                                   COMCIDP    3198    A
COMCIDP   ST=XEQ   =      ST=STP6                                         COMCIDP    3199    A
COMCIDP   ST=XFR   SPACE  4,10                                            COMCIDP    3200    A
COMCIDP  *         HERE TO PROCESS *XFER NR*.                             COMCIDP    3201    A
COMCIDP                                                                   COMCIDP    3202    A
COMCIDP   ST=XFR   ZR     X5,ST=XFR1  IF EOS ENCOUNTERED                  COMCIDP    3203    A
COMCIDP            RJ     PAS         PARSE SUBEXPRESSION -- NR           COMCIDP    3204    A
COMCIDP            NZ     X3,ST=XFR2  IF NR NOT NULL                      COMCIDP    3205    A
COMCIDP                                                                   COMCIDP    3206    A
COMCIDP   ST=XFR1  SX6    L.XFT       (X6) = DEFAULT NR = ALL ENTRIES     COMCIDP    3207    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP                                                                                COMCIDP     3208     A
COMCIDP   ST=XFR2  RJ     LXT         LIST XFER TABLE                                  COMCIDP     3209     A
COMCIDP            EQ     IDP=MN                                                       COMCIDP     3210     A
COMCIDP   ST=XNR   SPACE  4,10                                                         COMCIDP     3211     A
COMCIDP  *         HERE TO PROCESS *STNR,NR,L,RJ*.                                     COMCIDP     3212     A
COMCIDP                                                                                COMCIDP     3213     A
COMCIDP   ST=XNR   SX6    XTC.NR                                                       COMCIDP     3214     A
COMCIDP            SA6    IDPXTC                                                       COMCIDP     3215     A
COMCIDP            ZR     X5,ST=STP5  IF EOS ENCOUNTERED                               COMCIDP     3216     A
COMCIDP            RJ     PAS         PARSE SUBEXPRESSION-- NR                         COMCIDP     3217     A
COMCIDP            ZR     X3,ST=STP5  IF NR IS NULL                                    COMCIDP     3218     A
COMCIDP            SA6    IDPXA1                                                       COMCIDP     3219     A
COMCIDP            SA6    IDPXA2                                                       COMCIDP     3220     A
COMCIDP            EQ     ST=STP5     REJOIN NORMAL STEP PROCESSING FLOW...            COMCIDP     3221     A
COMCIDP   IDP      TITLE  IDP "IDPVER"/SUPPORTING SUBROUTINES.                         COMCIDP     3222     A
COMCIDP   ADZ      SPACE  4,10                                                         COMCIDP     3223     A
COMCIDP  **        ADZ - ADD A WORD TO IDP TABLE.                                      COMCIDP     3224     A
COMCIDP  *                                                                             COMCIDP     3225     A
COMCIDP  *                                                                             COMCIDP     3226     A
COMCIDP  *                THIS ROUTINE WILL ADD A SINGLE WORD TO AN IDP FIXED          COMCIDP     3227     A
COMCIDP  *         LENGTH TABLE THAT IS TERMINATED BY A ZERO WORD, AND WHERE           COMCIDP     3228     A
COMCIDP  *         AN AVAILABLE SPACE IS MARKED BY A -1.                               COMCIDP     3229     A
COMCIDP  *                                                                             COMCIDP     3230     A
COMCIDP  *         ENTRY  (X6)    =  WORD TO BE ADDED TO TABLE                         COMCIDP     3231     A
COMCIDP  *                (B6)    =  FWA OF TABLE                                      COMCIDP     3232     A
COMCIDP  *                                                                             COMCIDP     3233     A
COMCIDP  *         EXIT   TO *SE.TOV* IF TABLE OVERFLOW                                COMCIDP     3234     A
COMCIDP  *                (X1)    =  -1 IF ENTRY ALREADY IN TABLE, ELSE                COMCIDP     3235     A
COMCIDP  *                       =  ENTRY THAT WAS MADE IN TABLE                       COMCIDP     3236     A
COMCIDP  *                (B7)    =  ORDINAL OF ENTRY THAT WAS MADE OR OF MATCHING     COMCIDP     3237     A
COMCIDP  *                           ENTRY                                             COMCIDP     3238     A
COMCIDP  *                                                                             COMCIDP     3239     A
COMCIDP  *         USES   X - 0,1,2,3,6,7                                              COMCIDP     3240     A
COMCIDP  *                A - 1,2                                                      COMCIDP     3241     A
COMCIDP  *                B - 7                                                        COMCIDP     3242     A
COMCIDP  *                                                                             COMCIDP     3243     A
COMCIDP  *         CALLS  IDP.ER,PRIDP,SKT                                             COMCIDP     3244     A
COMCIDP                                                                                COMCIDP     3245     A
COMCIDP                                                                                COMCIDP     3246     A
COMCIDP   ADZ      SUBR              ** ENTRY/EXIT **                                  COMCIDP     3247     A
COMCIDP            BX1    X6                                                           COMCIDP     3248     A
COMCIDP            MX0    60                                                           COMCIDP     3249     A
COMCIDP            SA6    ADZA        SAVE (X6)                                        COMCIDP     3250     A
COMCIDP            RJ     SKT         SEARCH TO SEE IF ENTRY ALREADY IN TABLE          COMCIDP     3251     A
COMCIDP            SX1    -B1                                                          COMCIDP     3252     A
COMCIDP            NZ     X2,EXIT.    IF ENTRY ALREADY IN TABLE                        COMCIDP     3253     A
COMCIDP            RJ     SKT         SEARCH FOR AVAILABLE SPACE                       COMCIDP     3254     A
COMCIDP            ZR     X2,SE.TOV   IF NO AVAILABLE SPACE LEFT                       COMCIDP     3255     A
COMCIDP            SA1    ADZA                                                         COMCIDP     3256     A
COMCIDP            BX6    X1                                                           COMCIDP     3257     A
COMCIDP            SA6    A2          MAKE NEW TABLE ENTRY                             COMCIDP     3258     A
COMCIDP            EQ     EXIT.                                                        COMCIDP     3259     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP                                                                  COMCIDP     3260     A
COMCIDP                                                                  COMCIDP     3261     A
COMCIDP   ADZA     BSSZ   1                                              COMCIDP     3262     A
COMCIDP   BRK      SPACE  4,10                                           COMCIDP     3263     A
COMCIDP  **        BRK - BREAK PROCESSOR.                                COMCIDP     3264     A
COMCIDP  *                                                               COMCIDP     3265     A
COMCIDP  *                                                               COMCIDP     3266     A
COMCIDP  *         1. IF THIS IS AN ASSEMBLED BREAK (NOT *IDP* GENERATED),  COMCIDP     3267     A
COMCIDP  *            THEN CONTROL IS MERELY RETURNED TO CALLER.         COMCIDP     3268     A
COMCIDP  *                                                               COMCIDP     3269     A
COMCIDP  *         2. IF THIS IS AN *IDP* GENERATED BREAK, THE 2 PARALLEL TABLES,  COMCIDP     3270     A
COMCIDP  *            *IDPBA/IDPBC*, ARE QUERIED FOR INFORMATION THAT WILL BE  COMCIDP     3271     A
COMCIDP  *            PLUGGED INTO A DUMMY PARAMETER LIST FOR THIS BREAK. THE  COMCIDP     3272     A
COMCIDP  *            PREVIOUS CONTENTS OF THE BREAK ADDR ARE PICKED UP FROM  COMCIDP     3273     A
COMCIDP  *            *IDPBC* AND PLUGGED AT *BC=BRAD* SO THAT THEY CAN BE  COMCIDP     3274     A
COMCIDP  *            EXECUTED BEFORE EXITING *IDP*.                     COMCIDP     3275     A
COMCIDP  *                                                               COMCIDP     3276     A
COMCIDP  *         TABLE FORMATS ARE--                                   COMCIDP     3277     A
COMCIDP  *                                                               COMCIDP     3278     A
COMCIDP  *         IDPBA ENTRY AT (IDPBA+ORDINAL)                        COMCIDP     3279     A
COMCIDP  *                                                               COMCIDP     3280     A
COMCIDP  *         VFD    6/BREAK TYPE CODE (BTC)                        COMCIDP     3281     A
COMCIDP  *         VFD    18/ ADDR1                                      COMCIDP     3282     A
COMCIDP  *         VFD    18/ ADDR2   FOR A CONDITIONAL BREAK (BRNE,...) THE  COMCIDP     3283     A
COMCIDP  *                              CONTENTS OF THE CORE LOCATION ADDR1 WILL  COMCIDP     3284     A
COMCIDP  *                              BE COMPARED TO THE CONTENTS OF CORE  COMCIDP     3285     A
COMCIDP  *                              LOCATION ADDR2.                  COMCIDP     3286     A
COMCIDP  *         VFD    18/ BREAK ADDR                                 COMCIDP     3287     A
COMCIDP  *                                                               COMCIDP     3288     A
COMCIDP  *         IDPBC ENTRY AT (IDPBC+2*ORDINAL)                      COMCIDP     3289     A
COMCIDP  *                                                               COMCIDP     3290     A
COMCIDP  *         VFD    60/ PREVIOUS CONTENTS OF BREAK ADDR            COMCIDP     3291     A
COMCIDP  *         VFD    15/ LL                                         COMCIDP     3292     A
COMCIDP  *         VFD    15/ UL                                         COMCIDP     3293     A
COMCIDP  *         VFD    12/ INC                                        COMCIDP     3294     A
COMCIDP  *         VFD    18/ SNAP COUNT                                 COMCIDP     3295     A
COMCIDP  *                                                               COMCIDP     3296     A
COMCIDP  *            IF THIS IS A CONDITIONAL *IDP* GENERATED BREAK     COMCIDP     3297     A
COMCIDP  *            (BRPL,BREQ,...), THE RELATIONAL CONDITION IS EVALUATED.  COMCIDP     3298     A
COMCIDP  *            A TRUE RESULT MEANS THE BREAK WILL BE HONORED, A FALSE THAT  COMCIDP     3299     A
COMCIDP  *            IT WILL NOT.                                       COMCIDP     3300     A
COMCIDP  *                                                               COMCIDP     3301     A
COMCIDP  *         ENTRY  (A1,X1) = A+C OF IDP ENTRY POINT.              COMCIDP     3302     A
COMCIDP  *                                                               COMCIDP     3303     A
COMCIDP  *         EXIT   (X5) = FWA OF PARAMETER LIST, ELSE             COMCIDP     3304     A
COMCIDP  *                     = .ZR. IF NO BREAK THIS TIME.             COMCIDP     3305     A
COMCIDP  *                                                               COMCIDP     3306     A
COMCIDP  *                (ADR=RJ)  = ADDR OF *RJ IDP=* WORD.            COMCIDP     3307     A
COMCIDP  *                (FW=PARM) = FWA OF PARAMETER LIST (ACTUAL OR INVENTED).  COMCIDP     3308     A
COMCIDP  *                                                               COMCIDP     3309     A
COMCIDP  *                (IDPPREG) = PSEUDO P REGISTER, I.E. ADDR OF NEXT  COMCIDP     3310     A
COMCIDP  *                            INSTRUCTION TO XEQ.                COMCIDP     3311     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *                (IDPPOS)  = POS COUNTER, I.E. SET EQUAL TO 60D
COMCIDP  *                          (FORCED UPPER).
COMCIDP  *                (BC=BRAD) = SAVED CONTENTS OF IDP GENERATED BREAK WORD,
COMCIDP  *                          ELSE .ZR. IF NONE.
COMCIDP  *
COMCIDP  *         USES   ALL
COMCIDP  *
COMCIDP  *         CALLS  CBC,CHK,IEX,SKT
COMCIDP
COMCIDP
COMCIDP   BRK      SUBR               ** ENTRY/EXIT **
COMCIDP            SX6    60D
COMCIDP            SX7    0
COMCIDP            SA6    IDPPOS      SET UP POS COUNTER
COMCIDP            SA7    BC=BRAD     INITIALIZE TO *NO SAVED BREAK CONTENTS*
COMCIDP            RJ     IEX         INITIALIZE EXECUTIVE
COMCIDP
COMCIDP            SA1    ADR=RJ      (X1) = ADDR OF CALLING *RJ IDP=*
COMCIDP            SA5    FW=PARM     (X5) = FWA OF *BREAK* PARAMETER LIST
COMCIDP            SX6    X1+1
COMCIDP            SA6    IDPPREG     SET UP PSEUDO P REGISTER
COMCIDP            GT     B2,B0,BRK10 IF ASSEMBLED *BREAK* W/ PARAM LIST
COMCIDP
COMCIDP            BX7    -X7-X7      (X7) = -0
COMCIDP            MX0    -18
COMCIDP            SA7    AP=HDR      SET TO *IDP GENERATED BREAK HEADER*
COMCIDP            BX0    -X0
COMCIDP            MI     B2,EXIT.    IF ASSEMBLED *RJ IDP=* W/O PARM LIST
COMCIDP
COMCIDP            SB6    IDPBA
COMCIDP            RJ     SKT         SEARCH FOR BREAK ADDR
COMCIDP            ZR     X2,SE.BRK   IF NO FIND, IRRETRIEVABLE ERROR...
COMCIDP            SB6    B7+B7
COMCIDP            SA3    IDPBC+B6+1  (X3) = 15/LL,15/UL,12/INC,18/CNT
COMCIDP            SX6    X3
COMCIDP            BX7    -X0*X3      (X7) = 15/LL,15/UL,12/INC,18/0
COMCIDP            SA6    AP=CNT
COMCIDP            SX4    X3+B1       (X4) = CNT+1
COMCIDP            BX7    X7+X4       (X7) = 15/LL,15/UL,12/INC,18/CNT+1
COMCIDP            SA7    A3
COMCIDP            MX0    -15
COMCIDP            LX3    15
COMCIDP            BX6    -X0*X3      (X6) = LL
COMCIDP            LX3    15
COMCIDP            BX7    -X0*X3      (X7) = UL
COMCIDP            SA6    AP=LL
COMCIDP            SA7    A6+B1
COMCIDP            MX0    -12
COMCIDP            LX3    12
COMCIDP            BX6    -X0*X3      (X6) = INC
COMCIDP            SA6    A7+B1
COMCIDP
```

```
         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *         SET UP EXIT SO THAT PREVIOUS CONTENTS OF BREAK ADDR WILL BE      COMCIDP    3364    A
COMCIDP  *           EXECUTED BEFORE EXITING *IDP*.                                 COMCIDP    3365    A
COMCIDP  *                                                                          COMCIDP    3366    A
COMCIDP            SA3    A3-B1       (X3) = PREVIOUS CONTENTS OF BREAK ADDR        COMCIDP    3367    A
COMCIDP            BX6    X3                                                        COMCIDP    3368    A
COMCIDP            SA6    BC=BRAD                                                   COMCIDP    3369    A
COMCIDP                                                                             COMCIDP    3370    A
COMCIDP  *         CHECK TO SEE IF BREAK CONDITION SATISFIED.                       COMCIDP    3371    A
COMCIDP                                                                             COMCIDP    3372    A
COMCIDP            MX0    -6                                                        COMCIDP    3373    A
COMCIDP            LX2    6                                                         COMCIDP    3374    A
COMCIDP            BX6    -X0*X2      (X6) = BREAK TYPE CODE (BTC)                  COMCIDP    3375    A
COMCIDP            SB5    X6                                                        COMCIDP    3376    A
COMCIDP            MX0    -18                                                       COMCIDP    3377    A
COMCIDP            LX2    18                                                        COMCIDP    3378    A
COMCIDP            BX6    -X0*X2                                                    COMCIDP    3379    A
COMCIDP            SA1    X6          (X1) = (ADDR1)                                COMCIDP    3380    A
COMCIDP            LX2    18                                                        COMCIDP    3381    A
COMCIDP            BX6    -X0*X2                                                    COMCIDP    3382    A
COMCIDP            SA2    X6          (X2) = (ADDR2)                                COMCIDP    3383    A
COMCIDP            RJ     CBC         CHECK BREAK CONDITION                         COMCIDP    3384    A
COMCIDP            NZ     B6,EXIT.    IF CONDITION WAS SATISFIED (TRUE)             COMCIDP    3385    A
COMCIDP            MX5    0                                                         COMCIDP    3386    A
COMCIDP            EQ     EXIT.                                                     COMCIDP    3387    A
COMCIDP                                                                             COMCIDP    3388    A
COMCIDP  *         HERE IF ASSEMBLED *BREAK* WITH A PARAMETER LIST.                 COMCIDP    3389    A
COMCIDP  *                                                                          COMCIDP    3390    A
COMCIDP  *         NEED TO CHECK (X5)+SN=BRAD.                                      COMCIDP    3391    A
COMCIDP                                                                             COMCIDP    3392    A
COMCIDP   BRK10    SA1    X5+SN=BRAD  (X1) = NEW PSEUDO P REG                       COMCIDP    3393    A
COMCIDP            SB2    X1+                                                       COMCIDP    3394    A
COMCIDP            LE     B2,B1,EXIT. IF NEW PSEUDO P IS BAD, IGNORE IT...          COMCIDP    3395    A
COMCIDP            RJ     CHK         CHECK CM ADDRESS                              COMCIDP    3396    A
COMCIDP            MI     B2,EXIT.    IF NEW PSEUDO P IS BAD, IGNORE IT...          COMCIDP    3397    A
COMCIDP                                                                             COMCIDP    3398    A
COMCIDP            SX6    B2                                                        COMCIDP    3399    A
COMCIDP            SX7    B2-B1                                                     COMCIDP    3400    A
COMCIDP            SA6    IDPPREG     UPDATE PSEUDO P REG                           COMCIDP    3401    A
COMCIDP            SA7    ADR=RJ      UPDATE CALLING ADDR                           COMCIDP    3402    A
COMCIDP            EQ     EXIT.                                                     COMCIDP    3403    A
COMCIDP   .IDPOS   ENDIF                                                            CCGA018      12    A
COMCIDP   CAD      SPACE  4,10                                                      COMCIDP    3404    A
COMCIDP  **        CAD - CONVERT ADDRESS FROM BINARY TO DPC.                        COMCIDP    3405    A
COMCIDP  *                                                                          COMCIDP    3406    A
COMCIDP  *                                                                          COMCIDP    3407    A
COMCIDP  *         THIS ROUTINE CONVERTS A BINARY ADDRESS (18 BITS) TO DPC          COMCIDP    3408    A
COMCIDP  *         IN ONE OF 2 WAYS --                                              COMCIDP    3409    A
COMCIDP  *                                                                          COMCIDP    3410    A
COMCIDP  *           1. IF *FRA=* (FIND RELATIVE ADDRESS) IS DEFINED .AND.          COMCIDP    3411    A
COMCIDP  *              *IDPFLG/IDF.ADRP* INDICATES THAT WE ARE TO CONVERT          COMCIDP    3412    A
COMCIDP  *              THE ADDR TO DECK RELATIVE, THEN *CAD* WILL CALL *FRA=*      COMCIDP    3413    A
COMCIDP  *              TO FIND AND CONVERT THE RELATIVE ADDRESS.                   COMCIDP    3414    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *                                                                  COMCIDP    3415    A
COMCIDP  *          2. IF *FRA=* IS NOT DEFINED .OR. *IDPFLG/IDF.ADRP* INDICATES  COMCIDP    3416    A
COMCIDP  *             THAT WE ARE TO CONVERT THE ADDR TO AN ABSOLUTE ADDR,       COMCIDP    3417    A
COMCIDP  *             *CAD* WILL CALL *COD* (CONVERT BINARY TO OCTAL DPC)        COMCIDP    3418    A
COMCIDP  *             TO CONVERT THE ADDR.                                       COMCIDP    3419    A
COMCIDP  *                                                                        COMCIDP    3420    A
COMCIDP  *        ENTRY  (X1) = ADDR TO CONVERT, IN BINARY.                       COMCIDP    3421    A
COMCIDP  *                                                                        COMCIDP    3422    A
COMCIDP  *        EXIT   (X6) = ..NNNNNN..  IF ABSOLUTE CONVERSION                COMCIDP    3423    A
COMCIDP  *                    = .+NNNNNN..  IF RELATIVE CONVERSION                COMCIDP    3424    A
COMCIDP  *                                                                        COMCIDP    3425    A
COMCIDP  *                      WHERE . = BLANK(55B)                              COMCIDP    3426    A
COMCIDP  *                            N = OCTAL DIGIT (ADDR IS RIGHT JUSTIFIED    COMCIDP    3427    A
COMCIDP  *                                TO BIT 3*CHAR, LEADING 0 SUPPRESSION)   COMCIDP    3428    A
COMCIDP  *                            + = RELATIVE ADDR INDICATOR                 COMCIDP    3429    A
COMCIDP  *                                                                        COMCIDP    3430    A
COMCIDP  *        USES   ALL BUT A0,X0,A5,X5,A6   (INCLUDES ALL CALLS)            COMCIDP    3431    A
COMCIDP  *                                                                        COMCIDP    3432    A
COMCIDP  *        CALLS  COD,FRA                                                  COMCIDP    3433    A
COMCIDP                                                                           COMCIDP    3434    A
COMCIDP                                                                           COMCIDP    3435    A
COMCIDP   CAD     SUBR                ** ENTRY/EXIT **                            COMCIDP    3436    A
COMCIDP                                                                           COMCIDP    3437    A
COMCIDP  *        CHECK FOR AND CONVERT/FIND A RELATIVE ADDRESS.                  COMCIDP    3438    A
COMCIDP                                                                           COMCIDP    3439    A
COMCIDP           SA2    IDPFLG                                                   COMCIDP    3440    A
COMCIDP           LX2    59-IDF.ADRP                                              COMCIDP    3441    A
COMCIDP           PL     X2,CAD3     IF USER REQUESTED ABS CONVERSION             COMCIDP    3442    A
COMCIDP                                                                           COMCIDP    3443    A
COMCIDP           RJ     FRA         FIND RELATIVE ADDRESS                        COMCIDP    3444    A
COMCIDP           MI     B7,CAD3     IF USER DID NOT PROVIDE *FRA=*               COMCIDP    3445    A
COMCIDP                                                                           COMCIDP    3446    A
COMCIDP  *        SET UP EXIT CONDITIONS.                                         COMCIDP    3447    A
COMCIDP                                                                           COMCIDP    3448    A
COMCIDP           SA2    =R.   +.                                                 COMCIDP    3449    A
COMCIDP           MX3    6*CHAR                                                   COMCIDP    3450    A
COMCIDP           BX4    X3*X6       (X4) = NNNNNN0000  (.=BLANK(55B),0=00B)      COMCIDP    3451    A
COMCIDP           IX6    X4+X2       (X6) = NNNNNN...+  (.=BLANK(55B))            COMCIDP    3452    A
COMCIDP           LX6    -2*CHAR     (X6) = .+NNNNNN..                            COMCIDP    3453    A
COMCIDP           EQ     EXIT.                                                    COMCIDP    3454    A
COMCIDP                                                                           COMCIDP    3455    A
COMCIDP  *        HERE IF CONVERTING TO AN ABSOLUTE ADDRESS.                      COMCIDP    3456    A
COMCIDP                                                                           COMCIDP    3457    A
COMCIDP   CAD3    RJ     COD         CONVERT BINARY TO OCTAL DPC                  COMCIDP    3458    A
COMCIDP           LX6    2*CHAR      (X6) = ..NNNNNN..   (.=BLANK(55B))           COMCIDP    3459    A
COMCIDP           EQ     EXIT.                                                    COMCIDP    3460    A
COMCIDP   CBC     SPACE  4,10                                                     COMCIDP    3461    A
COMCIDP   .IDPOS  IFNE   .OS,2                                                    CCGA018      13    A
COMCIDP  **       CBC - CHECK BREAK CONDITION.                                    COMCIDP    3462    A
COMCIDP  *                                                                        COMCIDP    3463    A
COMCIDP  *                                                                        COMCIDP    3464    A
COMCIDP  *                THIS ROUTINE RETURNS A TRUE/FALSE VALUE FOR A GIVEN     COMCIDP    3465    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *         LOGICAL EXPRESSION-- C1.OP.C2                                COMCIDP    3466    A
COMCIDP  *                                                                      COMCIDP    3467    A
COMCIDP  *         ENTRY  (X1) = C1   1ST QUANTITY                              COMCIDP    3468    A
COMCIDP  *                (X2) = C2   2ND QUANTITY                              COMCIDP    3469    A
COMCIDP  *                (B5) = BREAK TYPE CODE (BTC)                          COMCIDP    3470    A
COMCIDP  *                                                                      COMCIDP    3471    A
COMCIDP  *         EXIT   (B6) = .NZ. IF TRUE, ELSE .ZR. IF FALSE               COMCIDP    3472    A
COMCIDP  *                                                                      COMCIDP    3473    A
COMCIDP  *         USES   X - 0,3,6,7                                           COMCIDP    3474    A
COMCIDP  *                A - NONE                                              COMCIDP    3475    A
COMCIDP  *                B - 6                                                 COMCIDP    3476    A
COMCIDP  *                                                                      COMCIDP    3477    A
COMCIDP  *         CALLS  NONE                                                  COMCIDP    3478    A
COMCIDP                                                                         COMCIDP    3479    A
COMCIDP                                                                         COMCIDP    3480    A
COMCIDP   CBC      SUBR                ** ENTRY/EXIT **                         COMCIDP    3481    A
COMCIDP            MI     B5,*+4S15    IF BTC IS BAD, AVOID SMELLY EVIL BUG...  COMCIDP    3482    A
COMCIDP            SB6    CBC2-CBC.OP (B6) = NR OF OPERATOR JP TABLE ENTRIES    COMCIDP    3483    A
COMCIDP            GT     B5,B6,*+4S15  IF BTC IS BAD, P U...DUMMKOPF           COMCIDP    3484    A
COMCIDP            SB6    B1           SET TO *CONDITION SATISFIED (TRUE)*      COMCIDP    3485    A
COMCIDP            ZR     B5,EXIT.     IF UNCONDITIONAL BREAK                  COMCIDP    3486    A
COMCIDP            BX3    X1-X2                                                 COMCIDP    3487    A
COMCIDP            IX6    X1-X2                                                 COMCIDP    3488    A
COMCIDP            IX7    X2-X1                                                 COMCIDP    3489    A
COMCIDP            MX0    0                                                     COMCIDP    3490    A
COMCIDP            IX6    X6+X0        REMOVE POSSIBLE -0                       COMCIDP    3491    A
COMCIDP            IX7    X7+X0                                                 COMCIDP    3492    A
COMCIDP            JP     CBC.OP+B5-1                                           COMCIDP    3493    A
COMCIDP                                                                         COMCIDP    3494    A
COMCIDP  *         CBC.OP - OPERATOR JUMP TABLE.                                COMCIDP    3495    A
COMCIDP                                                                         COMCIDP    3496    A
COMCIDP   CBC.OP   BSS    0                                                     COMCIDP    3497    A
COMCIDP            LOC    1                                                     COMCIDP    3498    A
COMCIDP                                                                         COMCIDP    3499    A
COMCIDP  *         HERE FOR .PL.C1                                              COMCIDP    3500    A
COMCIDP                                                                         COMCIDP    3501    A
COMCIDP   CBC.PL   PL     X1,EXIT.     IF TRUE                                  COMCIDP    3502    A
COMCIDP            EQ     CBC2                                                  COMCIDP    3503    A
COMCIDP                                                                         COMCIDP    3504    A
COMCIDP  *         HERE FOR .MI.C1                                              COMCIDP    3505    A
COMCIDP                                                                         COMCIDP    3506    A
COMCIDP   CBC.MI   MI     X1,EXIT.     IF TRUE                                  COMCIDP    3507    A
COMCIDP            EQ     CBC2                                                  COMCIDP    3508    A
COMCIDP                                                                         COMCIDP    3509    A
COMCIDP  *         HERE FOR .ZR.C1                                              COMCIDP    3510    A
COMCIDP                                                                         COMCIDP    3511    A
COMCIDP   CBC.ZR   ZR     X1,EXIT.     IF TRUE                                  COMCIDP    3512    A
COMCIDP            EQ     CBC2                                                  COMCIDP    3513    A
COMCIDP                                                                         COMCIDP    3514    A
COMCIDP  *         HERE FOR .NZ.C1                                              COMCIDP    3515    A
COMCIDP                                                                         COMCIDP    3516    A
COMCIDP   CBC.NZ   NZ     X1,EXIT.     IF TRUE                                  COMCIDP    3517    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            EQ     CBC2                                                    COMCIDP     3518     A
COMCIDP                                                                            COMCIDP     3519     A
COMCIDP  *         HERE FOR C1.EQ.C2                                               COMCIDP     3520     A
COMCIDP                                                                            COMCIDP     3521     A
COMCIDP   CBC.EQ   ZR     X3,EXIT.    IF TRUE                                     COMCIDP     3522     A
COMCIDP            EQ     CBC2                                                    COMCIDP     3523     A
COMCIDP                                                                            COMCIDP     3524     A
COMCIDP  *         HERE FOR C1.NE.C2                                               COMCIDP     3525     A
COMCIDP                                                                            COMCIDP     3526     A
COMCIDP   CBC.NE   NZ     X3,EXIT.    IF TRUE                                     COMCIDP     3527     A
COMCIDP            EQ     CBC2                                                    COMCIDP     3528     A
COMCIDP                                                                            COMCIDP     3529     A
COMCIDP  *         HERE FOR C1.LT.C2                                               COMCIDP     3530     A
COMCIDP                                                                            COMCIDP     3531     A
COMCIDP   CBC.LT   MI     X6,EXIT.    IF TRUE                                     COMCIDP     3532     A
COMCIDP            EQ     CBC2                                                    COMCIDP     3533     A
COMCIDP                                                                            COMCIDP     3534     A
COMCIDP  *         HERE FOR C1.GE.C2                                               COMCIDP     3535     A
COMCIDP                                                                            COMCIDP     3536     A
COMCIDP   CBC.GE   PL     X6,EXIT.    IF TRUE                                     COMCIDP     3537     A
COMCIDP            EQ     CBC2                                                    COMCIDP     3538     A
COMCIDP                                                                            COMCIDP     3539     A
COMCIDP  *         HERE FOR C1.LE.C2                                               COMCIDP     3540     A
COMCIDP                                                                            COMCIDP     3541     A
COMCIDP   CBC.LE   PL     X7,EXIT.    IF TRUE                                     COMCIDP     3542     A
COMCIDP            EQ     CBC2                                                    COMCIDP     3543     A
COMCIDP                                                                            COMCIDP     3544     A
COMCIDP  *         HERE FOR C1.GT.C2                                               COMCIDP     3545     A
COMCIDP                                                                            COMCIDP     3546     A
COMCIDP   CBC.GT   MI     X7,EXIT.    IF TRUE                                     COMCIDP     3547     A
COMCIDP            EQ     CBC2                                                    COMCIDP     3548     A
COMCIDP                                                                            COMCIDP     3549     A
COMCIDP   CBC.LOP  =      *           DEFINE LAST OP                              COMCIDP     3550     A
COMCIDP            LOC    *O                                                      COMCIDP     3551     A
COMCIDP                                                                            COMCIDP     3552     A
COMCIDP  *         HERE FOR CONDITION NOT SATISFIED (FALSE).                       COMCIDP     3553     A
COMCIDP                                                                            COMCIDP     3554     A
COMCIDP   CBC2     SB6    B0                                                      COMCIDP     3555     A
COMCIDP            EQ     EXIT.                                                   COMCIDP     3556     A
COMCIDP   CHK      SPACE  4,10                                                    COMCIDP     3557     A
COMCIDP   .IDPOS   ENDIF                                                           CCGA018       14     A
COMCIDP  **        CHK - CHECK CM ADDRESS.                                         COMCIDP     3558     A
COMCIDP  *                                                                         COMCIDP     3559     A
COMCIDP  *                                                                         COMCIDP     3560     A
COMCIDP  *         ENTRY  (B2)   =  ADDRESS TO BE CHECKED                          COMCIDP     3561     A
COMCIDP  *                                                                         COMCIDP     3562     A
COMCIDP  *         EXIT   (B2)   =  .MI. IF (B2) WAS BAD                           COMCIDP     3563     A
COMCIDP  *                       =  UNCHANGED IF OK                                COMCIDP     3564     A
COMCIDP  *                                                                         COMCIDP     3565     A
COMCIDP  *         USES   X - 1,2                                                  COMCIDP     3566     A
COMCIDP  *                A - 1                                                    COMCIDP     3567     A
COMCIDP  *                B - NONE    (USES B2 IF .GE. FL)                         COMCIDP     3568     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *                                                                    COMCIDP     3569     A
COMCIDP  *         CALLS  MEMORY                                              COMCIDP     3570     A
COMCIDP                                                                       COMCIDP     3571     A
COMCIDP                                                                       COMCIDP     3572     A
COMCIDP   CHK      SUBR                ** ENTRY/EXIT **                       COMCIDP     3573     A
COMCIDP                                                                       COMCIDP     3574     A
COMCIDP            SX2    =YCP.NFLS                                           COMCIDP     3575     A
COMCIDP            MI     B2,EXIT.    IF ADDR BAD TO START WITH              COMCIDP     3576     A
COMCIDP            MI     X2,CHK2     IF USER IS NOT DOING OWN MEMORY MGMT   COMCIDP     3577     A
COMCIDP            SA1    IDPFLG                                              COMCIDP     3578     A
COMCIDP            LX1    59-IDF.MEMP                                         COMCIDP     3579     A
COMCIDP            MI     X1,CHK3     IF USER HAS SELECTED *OPT MEM*         COMCIDP     3580     A
COMCIDP            SA1    X2          (X1) = CURRENT NOMINAL FL SCM          COMCIDP     3581     A
COMCIDP            EQ     CHK4                                                COMCIDP     3582     A
COMCIDP                                                                       COMCIDP     3583     A
COMCIDP   CHK2     SA1    RA.LWP                                              COMCIDP     3584     A
COMCIDP            MI     B2,EXIT.    IF ADDR IS BAD                         COMCIDP     3585     A
COMCIDP            SX1    X1          EXTEND SIGN BIT                        COMCIDP     3586     A
COMCIDP            PL     X1,CHK3     IF CMM NOT ACTIVE                      COMCIDP     3587     A
COMCIDP            BX1    -X1         (X1) = DABA                            COMCIDP     3588     A
COMCIDP            SA1    X1          (X1) = 42/STUFF,18/FL SCM              COMCIDP     3589     A
COMCIDP            SX2    B2                                                  COMCIDP     3590     A
COMCIDP            SX1    X1          (X1) = FL SCM                          COMCIDP     3591     A
COMCIDP            IX2    X2-X1       (X2) = ADDR - FL                       COMCIDP     3592     A
COMCIDP            MI     X2,EXIT.    IF ADDR LT FL                          COMCIDP     3593     A
COMCIDP                                                                       COMCIDP     3594     A
COMCIDP   CHK3     BX2    X6          SAVE X6 ACROSS MEMORY MACRO            COMCIDP     3595     A
COMCIDP            SX6    A6                                                  COMCIDP     3596     A
COMCIDP            SA6    CHKB        SAVE A6 ACROSS MEMORY MACRO            COMCIDP     3597     A
COMCIDP            MX6    0                                                   COMCIDP     3598     A
COMCIDP            SA6    CHKA        CLEAR MEM REQUEST WORD                 COMCIDP     3599     A
COMCIDP            MEMORY CM,A6,RCL   REQUEST CURRENT FIELD LENGTH           COMCIDP     3600          I
          -CCGA018
COMCIDP            MEMORY CM,CHKA,RCL    REQUEST CURRENT FIELD LENGTH        CCGA018       15     A
COMCIDP            SA1    CHKB        (X1) = SAVED A6                        COMCIDP     3601     A
COMCIDP            SA1    X1          (X1) = ((A6))                          COMCIDP     3602     A
COMCIDP            BX6    X1                                                  COMCIDP     3603     A
COMCIDP            SA6    A1          RESTORE A6                             COMCIDP     3604     A
COMCIDP            BX6    X2          RESTORE X6                             COMCIDP     3605     A
COMCIDP            SA1    CHKA        (X1) = 30/FL SCM,30/STUFF              COMCIDP     3606     A
COMCIDP            AX1    30D         (X1) = FL SCM                          COMCIDP     3607     A
COMCIDP                                                                       COMCIDP     3608     A
COMCIDP   CHK4     SX2    B2                                                  COMCIDP     3609     A
COMCIDP            IX2    X2-X1                                               COMCIDP     3610     A
COMCIDP            MI     X2,EXIT.    IF ADDR OK                             COMCIDP     3611     A
COMCIDP            SB2    -B2         SET TO *ADDR IS BAD*                   COMCIDP     3612     A
COMCIDP            EQ     EXIT.                                               COMCIDP     3613     A
COMCIDP                                                                       COMCIDP     3614     A
COMCIDP                                                                       COMCIDP     3615     A
COMCIDP   CHKA     BSSZ   1           REQUEST WORD FOR MEMORY MACRO          COMCIDP     3616     A
COMCIDP   CHKB     BSSZ   1           FOR SAVING A6 ACROSS MEMORY MACRO      COMCIDP     3617     A
COMCIDP   CIB      SPACE  4,10                                                COMCIDP     3618     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP   .IDPOS   IFNE   .OS,2                                          CCGA018      16     A
COMCIDP  **        CIB - CONVERT UNSIGNED INTEGER TO BINARY.            COMCIDP    3619     A
COMCIDP  *                                                              COMCIDP    3620     A
COMCIDP  *                                                              COMCIDP    3621     A
COMCIDP  *         *CIB* CONVERTS A STREAM OF *O.CONS* TOKENS TO A BINARY NUMBER  COMCIDP    3622     A
COMCIDP  *         VIA A CALLER-SPECIFIED BASE CONVERSION.              COMCIDP    3623     A
COMCIDP  *                                                              COMCIDP    3624     A
COMCIDP  *         ENTRY  (A5,X5) = A+C OF 1ST *O.CONS* TOKEN.          COMCIDP    3625     A
COMCIDP  *                (X1) = DEFAULT BASE.  (X1) CAN BE IN THE RANGE --  COMCIDP    3626     A
COMCIDP  *                                                              COMCIDP    3627     A
COMCIDP  *                        1 .LT. (X1) .LT 36D                   COMCIDP    3628     A
COMCIDP  *                                                              COMCIDP    3629     A
COMCIDP  *         EXIT   (B2) = .ZR. IF NO ERROR, ELSE .NZ.            COMCIDP    3630     A
COMCIDP  *                (X6) = BINARY NUMBER, ELSE .ZR. IF ERROR ((B2) .NZ.).  COMCIDP    3631     A
COMCIDP  *                (A5+1) = ADDR OF 1ST TOKEN FOLLOWING *CONS*.  COMCIDP    3632     A
COMCIDP  *                                                              COMCIDP    3633     A
COMCIDP  *         USES   X - 1,2,3,4,5,6,7                             COMCIDP    3634     A
COMCIDP  *                A - 1,2,5,6,7                                 COMCIDP    3635     A
COMCIDP  *                B - 2,3,4                                     COMCIDP    3636     A
COMCIDP  *                                                              COMCIDP    3637     A
COMCIDP  *         CALLS  MUL                                           COMCIDP    3638     A
COMCIDP                                                                 COMCIDP    3639     A
COMCIDP                                                                 COMCIDP    3640     A
COMCIDP   CIB      SUBR              ** ENTRY/EXIT **                   COMCIDP    3641     A
COMCIDP            SX6    X1                                            COMCIDP    3642     A
COMCIDP            MX7    0                                             COMCIDP    3643     A
COMCIDP            SA6    CIBA        SAVE (X1) = BASE                  COMCIDP    3644     A
COMCIDP            SA7    CIBC        SET TO *NO EXPLICIT BASE SO FAR*  COMCIDP    3645     A
COMCIDP            SB2    X5-O.CONS                                     COMCIDP    3646     A
COMCIDP            NZ     B2,SE.CONS  IF ERROR ON ENTRY                 COMCIDP    3647     A
COMCIDP                                                                 COMCIDP    3648     A
COMCIDP  *         CHECK FOR -B- (OCTAL) OR -D- (DECIMAL) SUFFIX.       COMCIDP    3649     A
COMCIDP                                                                 COMCIDP    3650     A
COMCIDP   CIB2     SA5    A5+1                                          COMCIDP    3651     A
COMCIDP            SX7    X5-O.CONS                                     COMCIDP    3652     A
COMCIDP            ZR     X7,CIB2     IF STILL *CONS* TOKEN             COMCIDP    3653     A
COMCIDP                                                                 COMCIDP    3654     A
COMCIDP            SB3    A5+         SAVE (A5) = LWA+1 OF *CONS*       COMCIDP    3655     A
COMCIDP            SB2    X5-O.VAR                                      COMCIDP    3656     A
COMCIDP            NZ     B2,CIB4     IF NOT *VAR*, NO -B/D- SUFFIX...  COMCIDP    3657     A
COMCIDP                                                                 COMCIDP    3658     A
COMCIDP            MX2    -CHAR                                         COMCIDP    3659     A
COMCIDP            MX3    TB.TOCL                                       COMCIDP    3660     A
COMCIDP            BX4    X3*X5       (X4) = CHARS IN *VAR*, -L- FMT    COMCIDP    3661     A
COMCIDP            LX5    CHAR                                          COMCIDP    3662     A
COMCIDP            BX3    -X2*X5      (X3) = 1ST CHAR IN *VAR*, -R- FMT COMCIDP    3663     A
COMCIDP            MX2    CHAR                                          COMCIDP    3664     A
COMCIDP            BX1    -X2*X4                                        COMCIDP    3665     A
COMCIDP            NZ     X1,CIB4     IF *VAR* LONGER THAN 1 CHAR, NO -B/D-...  COMCIDP    3666     A
COMCIDP            SB2    X3-1RB                                        COMCIDP    3667     A
COMCIDP            SX7    8           SET TO *OCTAL*                    COMCIDP    3668     A
COMCIDP            ZR     B2,CIB3     IF -B- SUFFIX                     COMCIDP    3669     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP                                                                                COMCIDP     3670     A
COMCIDP            SB2    X3-1RD                                                      COMCIDP     3671     A
COMCIDP            SX7    10D          SET TO *DECIMAL*                              COMCIDP     3672     A
COMCIDP            NZ     B2,CIB4      IF NO -D- SUFFIX                              COMCIDP     3673     A
COMCIDP                                                                                COMCIDP     3674     A
COMCIDP   CIB3     SA7    CIBA         SAVE EXPLICIT BASE                            COMCIDP     3675     A
COMCIDP            SA7    CIBC         SET TO *EXPLICIT BASE OCCURRED*               COMCIDP     3676     A
COMCIDP            SB3    B3+1                                                        COMCIDP     3677     A
COMCIDP                                                                                COMCIDP     3678     A
COMCIDP   *        CHECK BASE.                                                        COMCIDP     3679     A
COMCIDP                                                                                COMCIDP     3680     A
COMCIDP   CIB4     SA1    CIBA         (X1) = BASE                                   COMCIDP     3681     A
COMCIDP            SX2    X1-2                                                        COMCIDP     3682     A
COMCIDP            SX3    X1-36                                                       COMCIDP     3683     A
COMCIDP            MI     X2,CIB10     IF BASE .LT. 2, ERROR...                      COMCIDP     3684     A
COMCIDP            PL     X3,CIB10     IF BASE .GT. 35, ERROR...                     COMCIDP     3685     A
COMCIDP                                                                                COMCIDP     3686     A
COMCIDP   *        INITIALIZE FOR CONVERSION LOOP.                                    COMCIDP     3687     A
COMCIDP                                                                                COMCIDP     3688     A
COMCIDP            BX6    X6-X6        CLEAR (X6) = BINARY ACCUMULATOR               COMCIDP     3689     A
COMCIDP            SX7    B1                                                          COMCIDP     3690     A
COMCIDP            SA7    CIBB         INITIALIZE BASE**N ACCUMULATOR,               COMCIDP     3691     A
COMCIDP   *                             I.E. BASE**0 = 1                              COMCIDP     3692     A
COMCIDP            SB2    0            SET TO *NO ERROR SO FAR*                      COMCIDP     3693     A
COMCIDP            SB4    20D          (B4) = MAX NR OF DIGITS TO CONVERT            COMCIDP     3694     A
COMCIDP                                                                                COMCIDP     3695     A
COMCIDP   *        MAIN LOOP NODE FOR DIGIT-BY-DIGIT CONVERSION.                      COMCIDP     3696     A
COMCIDP   *                                                                            COMCIDP     3697     A
COMCIDP   *          1. CONVERT NEXT DIGIT TO BINARY, CHECKING FOR DIGIT              COMCIDP     3698     A
COMCIDP   *             VALIDITY ALONG THE WAY.                                        COMCIDP     3699     A
COMCIDP   *                                                                            COMCIDP     3700     A
COMCIDP   *          2. SCALE THIS DIGIT BY MULTIPLYING IT BY APPROPRIATE             COMCIDP     3701     A
COMCIDP   *             POWER OF BASE.                                                 COMCIDP     3702     A
COMCIDP   *                                                                            COMCIDP     3703     A
COMCIDP   *          3. ADD RESULTANT SCALED DIGIT INTO ACCUMULATOR.                   COMCIDP     3704     A
COMCIDP                                                                                COMCIDP     3705     A
COMCIDP   CIB5     SA5    A5-1                                                        COMCIDP     3706     A
COMCIDP            SX7    X5-O.CONS                                                   COMCIDP     3707     A
COMCIDP            NZ     X7,CIB11     IF NOT *CONS*, DONE...                        COMCIDP     3708     A
COMCIDP            MX4    TB.TOCL                                                     COMCIDP     3709     A
COMCIDP            BX5    X4*X5        (X5) = CHARS IN *CONS*, -L- FMT               COMCIDP     3710     A
COMCIDP                                                                                COMCIDP     3711     A
COMCIDP   CIB6     AX5    CHAR                                                        COMCIDP     3712     A
COMCIDP            MX4    -CHAR                                                       COMCIDP     3713     A
COMCIDP            ZR     X5,CIB5      IF (X5) EXHAUSTED                             COMCIDP     3714     A
COMCIDP            BX4    -X4*X5       (X4) = NEXT CHAR, -R- FMT                     COMCIDP     3715     A
COMCIDP            ZR     X4,CIB6      IF NULL (00B) CHARACTER, IGNORE...            COMCIDP     3716     A
COMCIDP            SB4    B4-B1                                                       COMCIDP     3717     A
COMCIDP            LT     B4,B0,CIB10 IF TRYING TO CONVERT TOO MANY DIGITS           COMCIDP     3718     A
COMCIDP            SX7    X4-1R9-1                                                    COMCIDP     3719     A
COMCIDP            PL     X7,CIB10     IF DIGIT .GT. 1R9                             COMCIDP     3720     A
COMCIDP            SX3    X4-1R0       (X3) = BINARY VALUE FOR THIS DIGIT            COMCIDP     3721     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            PL     X3,CIB7     IF 1R0 .LE. DIGIT .LE. 1R9                          COMCIDP     3722     A
COMCIDP            SX3    X4-1RA+10   (X3) = BINARY VALUE FOR THIS DIGIT                  COMCIDP     3723     A
COMCIDP                                                                                   COMCIDP     3724     A
COMCIDP   CIB7     SA1    CIBA        (X1) = BASE                                         COMCIDP     3725     A
COMCIDP            SA2    CIBB        (X2) = BASE**N ACCUMULATOR                          COMCIDP     3726     A
COMCIDP            IX4    X3-X1       (X4) = DIGIT - BASE                                 COMCIDP     3727     A
COMCIDP            MI     X4,CIB9     IF DIGIT .LT. BASE                                  COMCIDP     3728     A
COMCIDP                                                                                   COMCIDP     3729     A
COMCIDP   *        HERE IF DIGIT TOO BIG FOR BASE.                                         COMCIDP     3730     A
COMCIDP                                                                                   COMCIDP     3731     A
COMCIDP            SA3    CIBC        (X3) = .NZ. IF EXPLICIT BASE OCCURRED, ELSE .ZR.    COMCIDP     3732     A
COMCIDP            SX7    X1-8                                                            COMCIDP     3733     A
COMCIDP            NZ     X3,CIB10    IF EXPLICIT BASE, ERROR...                          COMCIDP     3734     A
COMCIDP            NZ     X7,CIB10    IF NOT BASE 8, ERROR...                             COMCIDP     3735     A
COMCIDP                                                                                   COMCIDP     3736     A
COMCIDP   *        HERE IF OCTAL WITH DIGIT .GE. 8.  TRY DECIMAL.                          COMCIDP     3737     A
COMCIDP                                                                                   COMCIDP     3738     A
COMCIDP            SX6    10D                                                             COMCIDP     3739     A
COMCIDP            SA6    CIBA        SET TO *DECIMAL CONVERSION*                         COMCIDP     3740     A
COMCIDP            SA5    B3                                                              COMCIDP     3741     A
COMCIDP            EQ     CIB4                                                            COMCIDP     3742     A
COMCIDP                                                                                   COMCIDP     3743     A
COMCIDP   *        HERE TO SCALE DIGIT.                                                    COMCIDP     3744     A
COMCIDP                                                                                   COMCIDP     3745     A
COMCIDP   CIB9     SX1    X3          (X1) = DIGIT                                        COMCIDP     3746     A
COMCIDP            RJ     MUL         INTEGER MULTIPLY (DIGIT*(BASE**N))                  COMCIDP     3747     A
COMCIDP            IX6    X6+X7                                                           COMCIDP     3748     A
COMCIDP            SA1    CIBA        (X1) = BASE                                         COMCIDP     3749     A
COMCIDP            SA2    CIBB        (X2) = BASE**N ACCUMULATOR                          COMCIDP     3750     A
COMCIDP            RJ     MUL         INTEGER MULTIPLY (BASE*(BASE**N))                   COMCIDP     3751     A
COMCIDP            SA7    A2          SAVE (X7) = BASE**(N+1)                             COMCIDP     3752     A
COMCIDP            EQ     CIB6                                                            COMCIDP     3753     A
COMCIDP                                                                                   COMCIDP     3754     A
COMCIDP   *        HERE IF AN ERROR.                                                       COMCIDP     3755     A
COMCIDP                                                                                   COMCIDP     3756     A
COMCIDP   CIB10    BX6    X6-X6                                                           COMCIDP     3757     A
COMCIDP            SB2    B1          SET TO *ERROR OCCURRED*                             COMCIDP     3758     A
COMCIDP                                                                                   COMCIDP     3759     A
COMCIDP   *        HERE WHEN READY TO EXIT.                                                COMCIDP     3760     A
COMCIDP                                                                                   COMCIDP     3761     A
COMCIDP   CIB11    SA5    B3-1        (A5) = ADDR OF LAST/FINAL TOKEN IN CONSTANT         COMCIDP     3762     A
COMCIDP            EQ     EXIT.                                                           COMCIDP     3763     A
COMCIDP                                                                                   COMCIDP     3764     A
COMCIDP                                                                                   COMCIDP     3765     A
COMCIDP   CIBA     BSSZ   1           SAVED BASE                                          COMCIDP     3766     A
COMCIDP   CIBB     BSSZ   1           SAVED BASE**N ACCUMULATOR                           COMCIDP     3767     A
COMCIDP   CIBC     BSSZ   1           .NZ. IF EXPLICIT BASE OCCURRED, ELSE .ZR.           COMCIDP     3768     A
COMCIDP   CLZ      SPACE  4,10                                                            COMCIDP     3769     A
COMCIDP   **       CLZ - CLEAR *IDP* TABLE.                                               COMCIDP     3770     A
COMCIDP   *                                                                               COMCIDP     3771     A
COMCIDP   *                                                                               COMCIDP     3772     A
COMCIDP   *                THIS ROUTINE WILL CLEAR AN *IDP* ZERO TERMINATED TABLE         COMCIDP     3773     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *         BY SETTING ALL ENTRIES TO *AVAILABLE*, I.E. -1.                COMCIDP    3774    A
COMCIDP  *                                                                        COMCIDP    3775    A
COMCIDP  *         ENTRY  (B6) = FWA OF TABLE TO BE CLEARED                       COMCIDP    3776    A
COMCIDP  *                                                                        COMCIDP    3777    A
COMCIDP  *         EXIT   NONE                                                    COMCIDP    3778    A
COMCIDP  *                                                                        COMCIDP    3779    A
COMCIDP  *         USES   X - 1,6                                                 COMCIDP    3780    A
COMCIDP  *                A - 1,6                                                 COMCIDP    3781    A
COMCIDP  *                B - NONE                                                COMCIDP    3782    A
COMCIDP  *                                                                        COMCIDP    3783    A
COMCIDP  *         CALLS  NONE                                                    COMCIDP    3784    A
COMCIDP                                                                           COMCIDP    3785    A
COMCIDP                                                                           COMCIDP    3786    A
COMCIDP   CLZ      SUBR              ** ENTRY/EXIT **                             COMCIDP    3787    A
COMCIDP            SA1    B6                                                      COMCIDP    3788    A
COMCIDP            MX6    -1                                                      COMCIDP    3789    A
COMCIDP                                                                           COMCIDP    3790    A
COMCIDP   CLZ2     ZR     X1,EXIT.    IF FINISHED                                 COMCIDP    3791    A
COMCIDP            SA6    A1                                                      COMCIDP    3792    A
COMCIDP            SA1    A1+B1                                                   COMCIDP    3793    A
COMCIDP            EQ     CLZ2                                                    COMCIDP    3794    A
COMCIDP   CON      SPACE  4,10                                                    COMCIDP    3795    A
COMCIDP  **        CON - CONNECT/DISCONT FILE.                                    COMCIDP    3796    A
COMCIDP  *                                                                        COMCIDP    3797    A
COMCIDP  *                                                                        COMCIDP    3798    A
COMCIDP  *         ENTRY  (A1,X1) = A + C OF 1ST WORD OF FET OF FILE              COMCIDP    3799    A
COMCIDP  *                (X2)    = +0  IF TO CONNECT                             COMCIDP    3800    A
COMCIDP  *                        =  1  IF TO DISCONT                             COMCIDP    3801    A
COMCIDP  *                                                                        COMCIDP    3802    A
COMCIDP  *         EXIT   SCOPE 3--                                               COMCIDP    3803    A
COMCIDP  *                (X1)    = .MI. IF PP ROUTINE *CON* DETECTED ERROR,      COMCIDP    3804    A
COMCIDP  *                                ELSE .PL.                               COMCIDP    3805    A
COMCIDP  *                KRONOS/NOS/SCOPE 2-- NONE                               COMCIDP    3806    A
COMCIDP  *                                                                        COMCIDP    3807    A
COMCIDP  *         USES   ALL BUT A0,A5,X5                                        COMCIDP    3808    A
COMCIDP  *                                                                        COMCIDP    3809    A
COMCIDP  *         CALLS  SCOPE 3-- SYSTEM                                        COMCIDP    3810    A
COMCIDP  *                KRONOS/NOS-- CLOSE,REQUEST,WRITER                       COMCIDP    3811    A
COMCIDP                                                                           COMCIDP    3812    A
COMCIDP                                                                           COMCIDP    3813    A
COMCIDP   CON      SUBR              ** ENTRY/EXIT **                             COMCIDP    3814    A
COMCIDP                                                                           COMCIDP    3815    A
COMCIDP   #OS2     IFEQ   .OS,2       IF SCOPE 2                                  COMCIDP    3816    A
COMCIDP            EQ     EXIT.                                                   COMCIDP    3817    A
COMCIDP   #OS2     ENDIF                                                          COMCIDP    3818    A
COMCIDP                                                                           COMCIDP    3819    A
COMCIDP   #OS3     IFEQ   .OS,3       IF SCOPE 3                                  COMCIDP    3820    A
COMCIDP            MX0    7*CHAR                                                  COMCIDP    3821    A
COMCIDP            BX6    X0*X1       (X6) = 42/0LFILENAME, 18/0                  COMCIDP    3822    A
COMCIDP            SA6    CONA                                                    COMCIDP    3823    A
COMCIDP            LX2    6                                                       COMCIDP    3824    A
COMCIDP            SYSTEM CON,RCL,CONA,X2                                         COMCIDP    3825    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP                                                                                COMCIDP     3826     A
COMCIDP            SA1    CONA                                                        COMCIDP     3827     A
COMCIDP            LX1    59-1                                                        COMCIDP     3828     A
COMCIDP            EQ     EXIT.                                                       COMCIDP     3829     A
COMCIDP                                                                                COMCIDP     3830     A
COMCIDP                                                                                COMCIDP     3831     A
COMCIDP   CONA     BSSZ   1           PARAMETER WORD FOR PP ROUTINE *CON*--           COMCIDP     3832     A
COMCIDP  *                              42/0LFILENAME                                  COMCIDP     3833     A
COMCIDP  *                              16/0, 1/ERROR FLAG, 1/COMPLETE BIT             COMCIDP     3834     A
COMCIDP                                                                                COMCIDP     3835     A
COMCIDP   #OS3     ELSE               IF KRONOS/NOS                                   COMCIDP     3836     A
COMCIDP            SX6    A1                                                          COMCIDP     3837     A
COMCIDP            SA6    CONA        SAVE (A1) = FWA OF FET                          COMCIDP     3838     A
COMCIDP            NZ     X2,CON2     IF TO DISCONT FILE                              COMCIDP     3839     A
COMCIDP                                                                                COMCIDP     3840     A
COMCIDP  *         HERE TO CONNECT FILE.                                              COMCIDP     3841     A
COMCIDP                                                                                COMCIDP     3842     A
COMCIDP            WRITER A1,,RCL     FLUSH BUFFER                                    COMCIDP     3843     A
COMCIDP            SA1    CONA                                                        COMCIDP     3844     A
COMCIDP            RETURN X1,RCL                                                      COMCIDP     3845     A
COMCIDP            SA1    CONA                                                        COMCIDP     3846     A
COMCIDP            SX6    2RTT                                                        COMCIDP     3847     A
COMCIDP            MX0    60-48                                                       COMCIDP     3848     A
COMCIDP            SA2    X1+B1       (X2) = 12/DEVICE TYPE, 48/OTHER                 COMCIDP     3849     A
COMCIDP            LX6    10*CHAR-2*CHAR                                              COMCIDP     3850     A
COMCIDP            BX2    -X0*X2                                                      COMCIDP     3851     A
COMCIDP            IX6    X2+X6       (X6) = 12/2RTT, 48/OTHER                        COMCIDP     3852     A
COMCIDP            SA6    A2                                                          COMCIDP     3853     A
COMCIDP            REQUEST X1,TT      ASSIGN FILE TO TERMINAL                         COMCIDP     3854     A
COMCIDP            EQ     EXIT.                                                       COMCIDP     3855     A
COMCIDP                                                                                COMCIDP     3856     A
COMCIDP  *         HERE TO DISCONT FILE.                                              COMCIDP     3857     A
COMCIDP                                                                                COMCIDP     3858     A
COMCIDP   CON2     WRITER A1,,RCL     FLUSH BUFFER                                    COMCIDP     3859     A
COMCIDP            SA1    CONA                                                        COMCIDP     3860     A
COMCIDP            RETURN X1,RCL                                                      COMCIDP     3861     A
COMCIDP            EQ     EXIT.                                                       COMCIDP     3862     A
COMCIDP                                                                                COMCIDP     3863     A
COMCIDP                                                                                COMCIDP     3864     A
COMCIDP   CONA     BSSZ   1           SAVED FWA OF FET                                COMCIDP     3865     A
COMCIDP   #OS3     ENDIF                                                              COMCIDP     3866     A
COMCIDP   CST      SPACE  4,10                                                        COMCIDP     3867     A
COMCIDP  **        CST - CLASSIFY STMT.                                               COMCIDP     3868     A
COMCIDP  *                                                                            COMCIDP     3869     A
COMCIDP  *                                                                            COMCIDP     3870     A
COMCIDP  *         *CST* PERFORMS THE TASK OF DETERMINING WHICH STMT                  COMCIDP     3871     A
COMCIDP  *         PROCESSOR IS TO PROCESS THE NEXT STMT IN THE TOKEN                 COMCIDP     3872     A
COMCIDP  *         BUFFER.                                                            COMCIDP     3873     A
COMCIDP  *                                                                            COMCIDP     3874     A
COMCIDP  *         ENTRY  (A5,X5) = A+C OF NEXT TOKEN TO TRY AND FIND A KEYWORD       COMCIDP     3875     A
COMCIDP  *                          MATCH FOR.                                        COMCIDP     3876     A
COMCIDP  *                                                                            COMCIDP     3877     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *         EXIT   (X6) = .NZ. IF ADDR OF FOUND *KEYW* ENTRY, ELSE      COMCIDP  3878  A
COMCIDP  *                     = .ZR. IF NO FIND.                             COMCIDP  3879  A
COMCIDP  *                                                                    COMCIDP  3880  A
COMCIDP  *                IF A FIND, (X6) .NZ. --                             COMCIDP  3881  A
COMCIDP  *                                                                    COMCIDP  3882  A
COMCIDP  *                (A5,X5) = A+C OF TOKEN IMMEDIATELY FOLLOWING KEYWORD. COMCIDP  3883  A
COMCIDP  *                                                                    COMCIDP  3884  A
COMCIDP  *         USES   ALL BUT A0                                          COMCIDP  3885  A
COMCIDP  *                                                                    COMCIDP  3886  A
COMCIDP  *         CALLS  CHK,SFN                                             COMCIDP  3887  A
COMCIDP                                                                       COMCIDP  3888  A
COMCIDP                                                                       COMCIDP  3889  A
COMCIDP   CST      SUBR              ** ENTRY/EXIT **                         COMCIDP  3890  A
COMCIDP                                                                       COMCIDP  3891  A
COMCIDP  *         INITIALIZATION.                                            COMCIDP  3892  A
COMCIDP  *                                                                    COMCIDP  3893  A
COMCIDP  *         WE FIRST NEED TO VERIFY THAT THE 1ST KEYWORD TABLE IS      COMCIDP  3894  A
COMCIDP  *         IN FACT AVAILABLE, AND IF NOT, WE MAKE SURE THAT AT        COMCIDP  3895  A
COMCIDP  *         LEAST IDP.S ORIGINAL KEYWORD TABLE (I.E. THE ONLY ONE      COMCIDP  3896  A
COMCIDP  *         WE HAPPEN TO KNOW THE LOCATION OF) IS MADE AVAILABLE       COMCIDP  3897  A
COMCIDP  *         SO THAT THE USER IS NOT LEFT WITH HIS OR HER PANTS DOWN.   COMCIDP  3898  A
COMCIDP                                                                       COMCIDP  3899  A
COMCIDP            SA1    FW=KEY      (X1) = FWA OF 1ST KEYWORD TABLE         COMCIDP  3900  A
COMCIDP            SB2    X1                                                  COMCIDP  3901  A
COMCIDP            SB4    X5          (B4) = TOKEN TYPE OF 1ST TOKEN TO CHECK COMCIDP  3902  A
COMCIDP            SB5    A5          SAVE ORIGINAL (A5)                      COMCIDP  3903  A
COMCIDP            SB6    X1                                                  COMCIDP  3904  A
COMCIDP            SB4    -B4         (BECAUSE THERE IS NO SX.I X.J-B.K)      COMCIDP  3905  A
COMCIDP            RJ     CHK         CHECK CM ADDR                           COMCIDP  3906  A
COMCIDP            PL     B2,CST3     IF 1ST KEYWORD TABLE IS AVAILABLE       COMCIDP  3907  A
COMCIDP            SX6    FW.KEY      (X6) = FWA OF IDP.S OWN KEYWORD TABLE   COMCIDP  3908  A
COMCIDP            SA6    FW=KEY                                              COMCIDP  3909  A
COMCIDP            EQ     SE.KEY      ** 1ST KEYWORD TABLE OUT-OF-RANGE **    COMCIDP  3910  A
COMCIDP                                                                       COMCIDP  3911  A
COMCIDP  *         HERE TO CHECK LINK TO NEXT KEYWORD TABLE.                  COMCIDP  3912  A
COMCIDP                                                                       COMCIDP  3913  A
COMCIDP   CST2     SA2    B6+B1       (X2) = LINK WORD                        COMCIDP  3914  A
COMCIDP            SB2    X2          (B2) = ADDR OF NEXT KEYWORD TABLE       COMCIDP  3915  A
COMCIDP            ZR     X2,CST7     IF END-OF-ALL KEYWORD TABLES, NO FIND...COMCIDP  3916  A
COMCIDP            RJ     CHK         CHECK CM ADDR                           COMCIDP  3917  A
COMCIDP            MI     B2,SE.KEY   IF NEXT KEYWORD TABLE IS OUT-OF-RANGE   COMCIDP  3918  A
COMCIDP            SB6    B2          (B6) = FWA OF NEXT KEYWORD TABLE        COMCIDP  3919  A
COMCIDP            BX1    X7*X5       RESTORE (X1) = CHARS TO SEARCH FOR, -L- FMT COMCIDP  3920  A
COMCIDP            EQ     CST4                                                COMCIDP  3921  A
COMCIDP                                                                       COMCIDP  3922  A
COMCIDP  *         KEYW TABLE SEARCH MAIN LOOP NODE.                          COMCIDP  3923  A
COMCIDP  *                                                                    COMCIDP  3924  A
COMCIDP  *         READY TO SCAN TOKEN(S) FOR A KEYWORD MATCH.                COMCIDP  3925  A
COMCIDP                                                                       COMCIDP  3926  A
COMCIDP   CST3     SA5    B5          (A5,X5) = A+C OF 1ST TOKEN TO SEARCH FOR COMCIDP  3927  A
COMCIDP            SX6    X5+B4                                               COMCIDP  3928  A
COMCIDP            NZ     X6,CST7     IF TOKEN TYPES DO NOT MATCH, NO FIND...COMCIDP  3929  A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            MX2    TB.TOCL                                          COMCIDP     3930     A
COMCIDP            BX1    X2*X5       (X1) = CHARS IN 1ST *VAR* TOKEN      COMCIDP     3931     A
COMCIDP            RJ     =XSFN       SPACE FILL NAME                      COMCIDP     3932     A
COMCIDP            BX1    X7*X6       (X1) = CHARS IN 1ST *VAR* TOKEN, -L- FMT  COMCIDP  3933     A
COMCIDP                                                                    COMCIDP     3934     A
COMCIDP  *         SEARCH KEYWORD TABLE FOR A MATCH OF CURRENT *VAR* TOKEN. COMCIDP     3935     A
COMCIDP                                                                    COMCIDP     3936     A
COMCIDP   CST4     SA2    B6          (X2) = KEYWORD TABLE ENTRY, WORD 1   COMCIDP     3937     A
COMCIDP            ZR     X2,CST2     IF END-OF-TABLE, TRY NEXT TABLE...   COMCIDP     3938     A
COMCIDP            LX2    0-KW.KEYP                                        COMCIDP     3939     A
COMCIDP            SA4    X2          (X4) = 1ST WORD OF KEYWORD LITERAL, -L- FMT  COMCIDP  3940     A
COMCIDP            SB6    B6+2                                             COMCIDP     3941     A
COMCIDP            LX2    0-KW.LENP+KW.KEYP-0                              COMCIDP     3942     A
COMCIDP            MX3    -KW.LENL                                         COMCIDP     3943     A
COMCIDP            BX6    -X3*X2      (X6) = NR OF BITS IN THIS KEYWORD    COMCIDP     3944     A
COMCIDP            SB7    X6                                               COMCIDP     3945     A
COMCIDP                                                                    COMCIDP     3946     A
COMCIDP   CST5     LE     B7,B0,CST3  IF NOTHING LEFT IN KEYWORD LITERAL   COMCIDP     3947     A
COMCIDP            BX6    X7*X4                                            COMCIDP     3948     A
COMCIDP            IX3    X1-X6                                            COMCIDP     3949     A
COMCIDP            NZ     X3,CST3     IF NO MATCH, TRY NEXT *KEYW* ENTRY...COMCIDP     3950     A
COMCIDP                                                                    COMCIDP     3951     A
COMCIDP  *         HERE TO SET UP FOR NEXT *VAR* TOKEN.                     COMCIDP     3952     A
COMCIDP                                                                    COMCIDP     3953     A
COMCIDP            SA5    A5+B1       (A5,X5) = A+C OF NEXT *VAR* TO SEARCH FOR  COMCIDP  3954     A
COMCIDP            MX2    TB.TOCL                                          COMCIDP     3955     A
COMCIDP            BX1    X2*X5       (X1) = CHARS IN NEXT *VAR* TOKEN, -L- FMT  COMCIDP  3956     A
COMCIDP            SX6    X5+B4                                            COMCIDP     3957     A
COMCIDP            NZ     X6,CST6     IF TOKEN TYPES DO NOT MATCH, FIND...COMCIDP     3958     A
COMCIDP            RJ     =XSFN       SPACE FILL NAME                      COMCIDP     3959     A
COMCIDP            SA4    A4+B1       (X4) = NEXT KEYWORD LITERAL, -L- FMT COMCIDP     3960     A
COMCIDP            BX1    X7*X6       (X1) = CHARS IN NEXT *VAR* TOKEN, -L- FMT  COMCIDP  3961     A
COMCIDP            SB7    B7-TB.TOCL  (B7) = NR OF BITS REMAINING IN THIS KEYWORD  COMCIDP  3962     A
COMCIDP            EQ     CST5                                             COMCIDP     3963     A
COMCIDP                                                                    COMCIDP     3964     A
COMCIDP  *         HERE IF WE HAVE A FIND.  SET UP EXIT CONDITIONS.         COMCIDP     3965     A
COMCIDP                                                                    COMCIDP     3966     A
COMCIDP   CST6     SX6    B6-2        (X6) = ADDR OF FOUND *KEYW* ENTRY    COMCIDP     3967     A
COMCIDP            EQ     EXIT.                                            COMCIDP     3968     A
COMCIDP                                                                    COMCIDP     3969     A
COMCIDP  *         HERE IF NO FIND.  SET UP EXIT CONDITIONS.                COMCIDP     3970     A
COMCIDP                                                                    COMCIDP     3971     A
COMCIDP   CST7     BX6    X6-X6       SET TO *NO FIND*                     COMCIDP     3972     A
COMCIDP            EQ     EXIT.                                            COMCIDP     3973     A
COMCIDP   .IDPOS   ENDIF                                                   CCGA018       17     A
COMCIDP   CXR      SPACE  4,10                                             COMCIDP     3974     A
COMCIDP  **        CXR - CHECK EXECUTIVE RJ.                                COMCIDP     3975     A
COMCIDP  *                                                                 COMCIDP     3976     A
COMCIDP  *                                                                 COMCIDP     3977     A
COMCIDP  *         *CXR* FIGURES OUT WHETHER THE *RJ* TO AN IDP EXECUTIVE   COMCIDP     3978     A
COMCIDP  *         HAS A PARAMETER LIST ASSOCIATED WITH IT.                 COMCIDP     3979     A
COMCIDP  *                                                                 COMCIDP     3980     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *         ENTRY  (X1) = ADDR OF EXECUTIVE                          COMCIDP   3981   A
COMCIDP  *                (X2) = ADDR OF CALLING *RJ*                       COMCIDP   3982   A
COMCIDP  *                                                                  COMCIDP   3983   A
COMCIDP  *         EXIT   (B2) = ADDR OF PARAMETER LIST, IF THERE IS ONE.   COMCIDP   3984   A
COMCIDP  *                     = .ZR. IF SPECIAL                            COMCIDP   3985   A
COMCIDP  *                              +  RJ  XEC                          COMCIDP   3986   A
COMCIDP  *                              -  VFD 30/0                         COMCIDP   3987   A
COMCIDP  *                     = .MI. IF NO PARAMETER LIST                  COMCIDP   3988   A
COMCIDP  *                                                                  COMCIDP   3989   A
COMCIDP  *         USES   X - 1,2,3,4,6,7                                   COMCIDP   3990   A
COMCIDP  *                A - 1,2,3,6                                       COMCIDP   3991   A
COMCIDP  *                B - 2                                             COMCIDP   3992   A
COMCIDP  *                                                                  COMCIDP   3993   A
COMCIDP  *         CALLS  CHK                                               COMCIDP   3994   A
COMCIDP                                                                     COMCIDP   3995   A
COMCIDP                                                                     COMCIDP   3996   A
COMCIDP   CXR      SUBR                ** ENTRY/EXIT **                     COMCIDP   3997   A
COMCIDP            SA3    CXRA         (X3) = *RJ-FORCED-UPPER* SKELETON    COMCIDP   3998   A
COMCIDP            LX1    30                                                COMCIDP   3999   A
COMCIDP            SX6    X2                                                COMCIDP   4000   A
COMCIDP            BX1    X1+X3        (X1) = 30/RJ XEC,30/0                COMCIDP   4001   A
COMCIDP            SA6    ADR=RJ                                            COMCIDP   4002   A
COMCIDP            SA2    X2           (A2,X2) = A+C OF WORD CONTAINING CALLING RJ  COMCIDP   4003   A
COMCIDP            IX7    X1-X2                                             COMCIDP   4004   A
COMCIDP            MX3    30                                                COMCIDP   4005   A
COMCIDP            SB2    B0                                                COMCIDP   4006   A
COMCIDP            BX4    X3*X2        (X4) = UPPER 30 BITS OF WORD CONTAINING RJ  COMCIDP   4007   A
COMCIDP            ZR     X7,EXIT.     IF SPECIAL 30/RJ XEC,30/0 FORM       COMCIDP   4008   A
COMCIDP            IX7    X1-X4                                             COMCIDP   4009   A
COMCIDP            SB2    -B1          SET TO *NO PARAMETER LIST*           COMCIDP   4010   A
COMCIDP            NZ     X7,EXIT.     IF NO PARAMETER LIST                 COMCIDP   4011   A
COMCIDP            BX4    -X3*X2       (X4) = LOWER 30 BITS OF WORD CONTAINING RJ  COMCIDP   4012   A
COMCIDP            SX2    X2           (X2) = POSSIBLE PARAMETER LIST ADDR  COMCIDP   4013   A
COMCIDP            IX7    X4-X2                                             COMCIDP   4014   A
COMCIDP            NZ     X7,EXIT.     IF LOWER 30 BITS CONTAIN MORE THAN AN ADDR  COMCIDP   4015   A
COMCIDP            SB2    X2                                                COMCIDP   4016   A
COMCIDP            RJ     CHK          CHECK CM ADDRESS                     COMCIDP   4017   A
COMCIDP            MI     B2,EXIT.     IF PARAMETER LIST ADDR IS BAD        COMCIDP   4018   A
COMCIDP            SA1    B2           (A1,X1) = A+C OF 1ST WORD OF PARAMETER LIST  COMCIDP   4019   A
COMCIDP            MX2    12                                                COMCIDP   4020   A
COMCIDP            BX7    X2*X1        (X7) = UPPER 12 BITS OF 1ST WORD OF PARM  COMCIDP   4021   A
COMCIDP            ZR     X7,EXIT.     IF PARAMETER LIST LOOKS OK           COMCIDP   4022   A
COMCIDP            SB2    -B1          SET TO *NO PARAMETER LIST*           COMCIDP   4023   A
COMCIDP            EQ     EXIT.                                             COMCIDP   4024   A
COMCIDP                                                                     COMCIDP   4025   A
COMCIDP                                                                     COMCIDP   4026   A
COMCIDP   CXRA     RJ     0                                                 COMCIDP   4027   A
COMCIDP   -        VFD    30/0                                              COMCIDP   4028   A
COMCIDP   DAB      SPACE  4,10                                              COMCIDP   4029   A
COMCIDP  **        DAB - DUMP AN -A- OR -B- REGISTER AND WHAT IT POINTS TO. COMCIDP   4030   A
COMCIDP  *                                                                  COMCIDP   4031   A
COMCIDP  *                                                                  COMCIDP   4032   A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *         ENTRY  (B4) = 0TR, WHERE                                COMCIDP    4033    A
COMCIDP  *                           T IS REGISTER TYPE                    COMCIDP    4034    A
COMCIDP  *                                 = 0 FOR -B- REGISTER            COMCIDP    4035    A
COMCIDP  *                                 = 1 FOR -A- REGISTER            COMCIDP    4036    A
COMCIDP  *                           R IS REGISTER NR (0-7)                COMCIDP    4037    A
COMCIDP  *                           E.G.  A6 WOULD BE 016                 COMCIDP    4038    A
COMCIDP  *                                                                 COMCIDP    4039    A
COMCIDP  *         EXIT   (B4) IS UNCHANGED                                COMCIDP    4040    A
COMCIDP  *                                                                 COMCIDP    4041    A
COMCIDP  *         USES   ALL BUT B4                                       COMCIDP    4042    A
COMCIDP  *                                                                 COMCIDP    4043    A
COMCIDP  *         CALLS  FAB,PRIDP                                        COMCIDP    4044    A
COMCIDP                                                                    COMCIDP    4045    A
COMCIDP                                                                    COMCIDP    4046    A
COMCIDP   DAB      SUBR                 ** ENTRY/EXIT **                   COMCIDP    4047    A
COMCIDP            SB6    SNAPLNE                                          COMCIDP    4048    A
COMCIDP            RJ     FAB          FORMAT -A- OR -B- REGISTER          COMCIDP    4049    A
COMCIDP            PRIDP  SNAPLNE                                          COMCIDP    4050    A
COMCIDP            EQ     EXIT.                                            COMCIDP    4051    A
COMCIDP   DAR      SPACE  4,10                                             COMCIDP    4052    A
COMCIDP  **        DAR - DUMP ALL REGISTERS.                               COMCIDP    4053    A
COMCIDP  *                                                                 COMCIDP    4054    A
COMCIDP  *                                                                 COMCIDP    4055    A
COMCIDP  *         ENTRY  NONE                                             COMCIDP    4056    A
COMCIDP  *                                                                 COMCIDP    4057    A
COMCIDP  *         EXIT   NONE                                             COMCIDP    4058    A
COMCIDP  *                                                                 COMCIDP    4059    A
COMCIDP  *         USES   ALL                                              COMCIDP    4060    A
COMCIDP  *                                                                 COMCIDP    4061    A
COMCIDP  *         CALLS  DAB,DUX,PRIDP,WOD                                COMCIDP    4062    A
COMCIDP                                                                    COMCIDP    4063    A
COMCIDP                                                                    COMCIDP    4064    A
COMCIDP   DAR      SUBR   0             ** ENTRY/EXIT **                   COMCIDP    4065    A
COMCIDP            SB4    B0           INITIALIZE (B4) = REGISTER COUNT, 0TR COMCIDP    4066    A
COMCIDP            PRIDP  ,,L.PRB                                          COMCIDP    4067    A
COMCIDP                                                                    COMCIDP    4068    A
COMCIDP            SA1    IDPFLG                                           COMCIDP    4069    A
COMCIDP            LX1    59-IDF.IDOP                                      COMCIDP    4070    A
COMCIDP            PL     X1,DAR5      IF NOT WRITING INTERACTIVELY,       COMCIDP    4071    A
COMCIDP  *                                PACK -A- AND -B- REGISTERS...    COMCIDP    4072    A
COMCIDP                                                                    COMCIDP    4073    A
COMCIDP  *         DUMP -B- REGISTERS.                                     COMCIDP    4074    A
COMCIDP                                                                    COMCIDP    4075    A
COMCIDP   DAR2     RJ     DAB          DUMP -B- REGISTER                   COMCIDP    4076    A
COMCIDP            SB4    B4+B1                                            COMCIDP    4077    A
COMCIDP            SB7    007B                                             COMCIDP    4078    A
COMCIDP            LE     B4,B7,DAR2   IF NOT DONE                         COMCIDP    4079    A
COMCIDP                                                                    COMCIDP    4080    A
COMCIDP            PRIDP  ,,L.PRB                                          COMCIDP    4081    A
COMCIDP                                                                    COMCIDP    4082    A
COMCIDP  *         DUMP -A- REGISTERS.                                     COMCIDP    4083    A
COMCIDP                                                                    COMCIDP    4084    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP   DAR3     RJ     DAB          DUMP -A- REGISTER                          COMCIDP     4085     A
COMCIDP            SB4    B4+B1                                                   COMCIDP     4086     A
COMCIDP            SB7    017B                                                    COMCIDP     4087     A
COMCIDP            LE     B4,B7,DAR3   IF NOT DONE                                COMCIDP     4088     A
COMCIDP                                                                            COMCIDP     4089     A
COMCIDP            PRIDP  ,,L.PRB                                                 COMCIDP     4090     A
COMCIDP                                                                            COMCIDP     4091     A
COMCIDP   *        DUMP -X- REGISTERS.                                             COMCIDP     4092     A
COMCIDP                                                                            COMCIDP     4093     A
COMCIDP   DAR4     RJ     DUX          DUMP -X- REGISTER                          COMCIDP     4094     A
COMCIDP            SB4    B4+B1                                                   COMCIDP     4095     A
COMCIDP            SB7    027B                                                    COMCIDP     4096     A
COMCIDP            LE     B4,B7,DAR4   IF NOT DONE                                COMCIDP     4097     A
COMCIDP            EQ     EXIT.                                                   COMCIDP     4098     A
COMCIDP                                                                            COMCIDP     4099     A
COMCIDP   *        HERE IF PACKING -A- AND -B- REGISTERS IN BATCH MODE.            COMCIDP     4100     A
COMCIDP                                                                            COMCIDP     4101     A
COMCIDP   DAR5     SB6    SNAPLNE                                                 COMCIDP     4102     A
COMCIDP            RJ     FAB          FORMAT -B- REGISTER                        COMCIDP     4103     A
COMCIDP            SB4    B4+10B                                                  COMCIDP     4104     A
COMCIDP            SB6    A6                                                      COMCIDP     4105     A
COMCIDP            RJ     FAB          FORMAT -A- REGISTER                        COMCIDP     4106     A
COMCIDP            PRIDP  SNAPLNE                                                 COMCIDP     4107     A
COMCIDP            SB4    B4-10B+1                                                COMCIDP     4108     A
COMCIDP            SB7    007B                                                    COMCIDP     4109     A
COMCIDP            LE     B4,B7,DAR5   IF NOT DONE                                COMCIDP     4110     A
COMCIDP                                                                            COMCIDP     4111     A
COMCIDP            PRIDP  ,,L.PRB                                                 COMCIDP     4112     A
COMCIDP            SB4    020B         (B4) = 0TR, WHERE T=2 (X REG)              COMCIDP     4113     A
COMCIDP            EQ     DAR4                                                    COMCIDP     4114     A
COMCIDP   DAZ      SPACE  4,10                                                    COMCIDP     4115     A
COMCIDP   .IDPOS   IFNE   .OS,2                                                   CCGA018       18     A
COMCIDP   **       DAZ - DEASSEMBLER.                                              COMCIDP     4116     A
COMCIDP   *                                                                        COMCIDP     4117     A
COMCIDP   *                                                                        COMCIDP     4118     A
COMCIDP   *                DEASSEMBLES A SINGLE BINARY INSTRUCTION TO ITS          COMCIDP     4119     A
COMCIDP   *        *COMPASS* MNEMONIC AND LISTS IT.                                COMCIDP     4120     A
COMCIDP   *                                                                        COMCIDP     4121     A
COMCIDP   *        ENTRY  (A5,     =  ADDR OF WORD CONTAINING INSTRUCTION TO       COMCIDP     4122     A
COMCIDP   *                             DEASSEMBLE                                 COMCIDP     4123     A
COMCIDP   *                   X5)  =  WORD CONTAINING INSTRUCTION TO DEASSEMBLE    COMCIDP     4124     A
COMCIDP   *                           NOTE-- ((A5)) MIGHT NOT BE EQUAL TO (X5),    COMCIDP     4125     A
COMCIDP   *                             AS WOULD BE THE CASE WHEN DASSEMBLING      COMCIDP     4126     A
COMCIDP   *                             THE SAVED PREVIOUS CONTENTS OF AN IDP      COMCIDP     4127     A
COMCIDP   *                             GENERATED BREAK (SEE *STP*)                COMCIDP     4128     A
COMCIDP   *                (B4)     =  POSITION COUNTER. LEFT MOST BIT TO BE       COMCIDP     4129     A
COMCIDP   *                             DEASSEMBLED IS BIT (B4)-1 IN (X5).         COMCIDP     4130     A
COMCIDP   *                             (BITS ARE NUMBERED 59 THRU 0) COMPASS      COMCIDP     4131     A
COMCIDP   *                             EQUIVALENT WOULD BE--                      COMCIDP     4132     A
COMCIDP   *                                 POS   (B4)    IN (X5)                  COMCIDP     4133     A
COMCIDP   *                (IDPFAD) =  .NZ. IF *DAZ* IS TO UNCONDITIONALLY FORCE  COMCIDP     4134     A
COMCIDP   *                                 OUT ADDR OF DEASSEMBLED INSTRUCTION,   COMCIDP     4135     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *                        =  .ZR. IF ADDR IS TO APPEAR ONLY IF INSTR IS      COMCIDP  4136  A
COMCIDP  *                                FORCED UPPER IN WORD, I.E. (B4)=60D        COMCIDP  4137  A
COMCIDP  *                                                                           COMCIDP  4138  A
COMCIDP  *         EXIT   (A5,X5)  =  UNCHANGED                                      COMCIDP  4139  A
COMCIDP  *                (B4)     =  UNCHANGED                                      COMCIDP  4140  A
COMCIDP  *                (X1)     =  .MI. IF INSTRUCTION DEASSEMBLED FORCES         COMCIDP  4141  A
COMCIDP  *                             UPPER, ELSE .PL.                              COMCIDP  4142  A
COMCIDP  *                (B5)     =  NR OF BITS DEASSEMBLED IF NO FORCE UPPER,      COMCIDP  4143  A
COMCIDP  *                         =  NR OF BITS IN INSTRUCTION THAT FORCED UPPER   COMCIDP  4144  A
COMCIDP  *                (IDPFAD) =  +0                                             COMCIDP  4145  A
COMCIDP  *                                                                           COMCIDP  4146  A
COMCIDP  *         USES   ALL BUT A5,X5,B4                                           COMCIDP  4147  A
COMCIDP  *                                                                           COMCIDP  4148  A
COMCIDP  *         LOCKED (A6+1)    =  NEXT AVAILABLE *SNAPLNE* LOCATION             COMCIDP  4149  A
COMCIDP  *                                                                           COMCIDP  4150  A
COMCIDP  *         CALLS  CAD,PRIDP,VFD                                              COMCIDP  4151  A
COMCIDP                                                                              COMCIDP  4152  A
COMCIDP                                                                              COMCIDP  4153  A
COMCIDP   DAZ      SUBR                ** ENTRY/EXIT **                              COMCIDP  4154  A
COMCIDP            SB6    60                                                         COMCIDP  4155  A
COMCIDP            LE     B4,B0,*+1S17  IF POSITION COUNTER IS BAD                  COMCIDP  4156  A
COMCIDP            SA1    =10H                                                       COMCIDP  4157  A
COMCIDP            SA2    IDPFAD                                                     COMCIDP  4158  A
COMCIDP            MX7    0                                                          COMCIDP  4159  A
COMCIDP            BX6    X1                                                         COMCIDP  4160  A
COMCIDP            SA7    A2          CLEAR *FORCE OUT ADDR UNCONDITIONALLY* FLAG    COMCIDP  4161  A
COMCIDP            NZ     X2,DAZ1     IF FORCING OUT ADDR UNCONDITIONALLY            COMCIDP  4162  A
COMCIDP            LT     B4,B6,DAZ2  IF THIS INSTRUCTION NOT UPPER                  COMCIDP  4163  A
COMCIDP                                                                              COMCIDP  4164  A
COMCIDP  *         HERE IF INSTRUCTION IS UPPER, OUTPUT ADDRESS.                     COMCIDP  4165  A
COMCIDP                                                                              COMCIDP  4166  A
COMCIDP   DAZ1     SX0    B4          SAVE (B4)                                      COMCIDP  4167  A
COMCIDP            SX1    A5          (X1) = ADDR OF INSTRUCTION TO DEASSEMBLE       COMCIDP  4168  A
COMCIDP            RJ     CAD         CONVERT ADDR TO DPC                            COMCIDP  4169  A
COMCIDP            SB4    X0+         RESTORE (B4)                                   COMCIDP  4170  A
COMCIDP                                                                              COMCIDP  4171  A
COMCIDP  *         BREAK OUT OP CODE.                                                COMCIDP  4172  A
COMCIDP                                                                              COMCIDP  4173  A
COMCIDP   DAZ2     SA6    SNAPLNE                                                    COMCIDP  4174  A
COMCIDP            SB6    -B4                                                        COMCIDP  4175  A
COMCIDP            MX0    -6                                                         COMCIDP  4176  A
COMCIDP            SB6    60+B6+6     60-(B4) = LEFT JUST SHIFT COUNT FOR INSTRTN    COMCIDP  4177  A
COMCIDP            LX5    X5,B6       (X5) = 54/OTHER STUFF, 6/OP CODE               COMCIDP  4178  A
COMCIDP            BX7    -X0*X5                                                     COMCIDP  4179  A
COMCIDP            LX5    3           (X5) = 51/OTHER STUFF,6/OP CODE,3/I            COMCIDP  4180  A
COMCIDP            MX0    -3                                                         COMCIDP  4181  A
COMCIDP            SA7    DAZ=OP                                                     COMCIDP  4182  A
COMCIDP            ZR     X7,DAZ12    IF 00B OP CODE, REST OF WORD IS DATA           COMCIDP  4183  A
COMCIDP                                                                              COMCIDP  4184  A
COMCIDP  *         BREAK OUT *I* FIELD.                                              COMCIDP  4185  A
COMCIDP                                                                              COMCIDP  4186  A
COMCIDP            BX7    -X0*X5                                                     COMCIDP  4187  A
```

```
         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP          LX5    3           (X5) = 48/OTHER STUFF,6/OP,3/I,3/J          COMCIDP    4188    A
COMCIDP          SX7    X7+1R0                                                  COMCIDP    4189    A
COMCIDP          SA7    A7+B1       SAVE (X7) = 1R<I>                           COMCIDP    4190    A
COMCIDP                                                                         COMCIDP    4191    A
COMCIDP  *       BREAK OUT *J* FIELD.                                            COMCIDP    4192    A
COMCIDP                                                                         COMCIDP    4193    A
COMCIDP          BX7    -X0*X5                                                  COMCIDP    4194    A
COMCIDP          LX5    3           (X5) = 45/OTHER,6/OP,3/I,3/J,3/K            COMCIDP    4195    A
COMCIDP          SX7    X7+1R0                                                  COMCIDP    4196    A
COMCIDP          SA7    A7+B1       SAVE (X7) = 1R<J>                           COMCIDP    4197    A
COMCIDP                                                                         COMCIDP    4198    A
COMCIDP  *       BREAK OUT *K* FIELD.                                            COMCIDP    4199    A
COMCIDP                                                                         COMCIDP    4200    A
COMCIDP          BX7    -X0*X5                                                  COMCIDP    4201    A
COMCIDP          LX5    18-3        (X5) = 30/OTHER,6/OP,3/I,3/J,18/Q           COMCIDP    4202    A
COMCIDP          SX7    X7+1R0                                                  COMCIDP    4203    A
COMCIDP          MX0    -18                                                     COMCIDP    4204    A
COMCIDP          SA7    A7+B1       SAVE (X7) = 1R<K>                           COMCIDP    4205    A
COMCIDP                                                                         COMCIDP    4206    A
COMCIDP  *       BREAK OUT *Q* 18 BIT ADDRESS FIELD.                             COMCIDP    4207    A
COMCIDP                                                                         COMCIDP    4208    A
COMCIDP          BX1    -X0*X5      (X1) = 42/0,18/Q                            COMCIDP    4209    A
COMCIDP          SB5    B4          SAVE (B4)                                   COMCIDP    4210    A
COMCIDP          SB7    60-60+B4-6-3-3-3-15  (B7) = 60-((60-(B4))+6+3+3+3+15)   COMCIDP    4211    A
COMCIDP          SX7    X1                                                      COMCIDP    4212    A
COMCIDP          LX5    X5,B7       RESTORE (X5)                                COMCIDP    4213    A
COMCIDP          SA7    DAZ=ADR                                                 COMCIDP    4214    A
COMCIDP          RJ     =XCOD       CONVERT BINARY TO OCTAL DPC-- *Q*           COMCIDP    4215    A
COMCIDP          SB4    B5          RESTORE (B4)                                COMCIDP    4216    A
COMCIDP          SB2    B2-B1                                                   COMCIDP    4217    A
COMCIDP          MX0    1                                                       COMCIDP    4218    A
COMCIDP          AX0    X0,B2                                                   COMCIDP    4219    A
COMCIDP          BX7    X0*X4       (X7) = 0L<Q>                                COMCIDP    4220    A
COMCIDP          SA7    A7+B1                                                   COMCIDP    4221    A
COMCIDP                                                                         COMCIDP    4222    A
COMCIDP  *       FETCH INSTRUCTION SKELETON AND CHECK FOR LENGTH CONFLICT.       COMCIDP    4223    A
COMCIDP                                                                         COMCIDP    4224    A
COMCIDP          SA1    DAZ=OP                                                  COMCIDP    4225    A
COMCIDP          SA2    DAZ=PS+X1   (X2) = INSTRUCTION SKELETON                 COMCIDP    4226    A
COMCIDP          SB5    15                                                      COMCIDP    4227    A
COMCIDP          BX1    X5                                                      COMCIDP    4228    A
COMCIDP          PL     X2,DAZ3     IF SHORT (15 BIT) INSTRUCTION               COMCIDP    4229    A
COMCIDP          SB5    B5+B5       (B5) = 30                                   COMCIDP    4230    A
COMCIDP                                                                         COMCIDP    4231    A
COMCIDP   DAZ3   LT     B4,B5,DAZ12 IF NOT ENOUGH ROOM FOR 30 BIT INSTR,        COMCIDP    4232    A
COMCIDP  *                                  MUST BE DATA                        COMCIDP    4233    A
COMCIDP                                                                         COMCIDP    4234    A
COMCIDP  *       CONVERT BINARY INSTRUCTION PARCEL TO OCTAL DPC.                 COMCIDP    4235    A
COMCIDP                                                                         COMCIDP    4236    A
COMCIDP          RJ     VFD         VARIABLE FIELD DEFINITION                   COMCIDP    4237    A
COMCIDP          SA6    A6+B1                                                   COMCIDP    4238    A
COMCIDP          BX6    X7                                                      COMCIDP    4239    A

                 0         1         2         3         4         5         6         7         8
                 12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            SA6    A6+1                                                    COMCIDP    4240    A
COMCIDP                                                                            COMCIDP    4241    A
COMCIDP  *         CHECK FOR X REGISTER BRANCH INSTRUCTION.                        COMCIDP    4242    A
COMCIDP                                                                            COMCIDP    4243    A
COMCIDP            SA1    DAZ=OP                                                  COMCIDP    4244    A
COMCIDP            SA2    DAZ=PS+X1    (X2) = INSTRUCTION SKELETON                COMCIDP    4245    A
COMCIDP            SX6    X1-03B                                                  COMCIDP    4246    A
COMCIDP            NZ     X6,DAZ4      IF NOT X REGISTER BRANCH                   COMCIDP    4247    A
COMCIDP            SA1    DAZ=I                                                   COMCIDP    4248    A
COMCIDP            SA2    DAZ=XJP+X1-1R0     (X2) = X REG BRANCH SKELETON          COMCIDP    4249    A
COMCIDP                                                                            COMCIDP    4250    A
COMCIDP  *         GENERATE ...OPI....      (.=BLANK(55B))                        COMCIDP    4251    A
COMCIDP                                                                            COMCIDP    4252    A
COMCIDP   DAZ4     SA1    DAZ=OPI      (X1) = ...000....   (0=00B)                COMCIDP    4253    A
COMCIDP            MX0    -2*CHAR                                                 COMCIDP    4254    A
COMCIDP            BX6    -X0*X2       (X6) = 2R<OP>                             COMCIDP    4255    A
COMCIDP            LX6    7*CHAR-2*CHAR                                           COMCIDP    4256    A
COMCIDP            MX0    -4                                                      COMCIDP    4257    A
COMCIDP            LX2    3+4          (X2) = 56/OTHER STUFF,4/1ST DESCRIPTOR    COMCIDP    4258    A
COMCIDP            BX3    -X0*X2                                                  COMCIDP    4259    A
COMCIDP            SA4    DAZ=I-1+X3                                              COMCIDP    4260    A
COMCIDP            BX6    X1+X6        (X6) = ...OP0....   (0=00B)                COMCIDP    4261    A
COMCIDP            LX4    5*CHAR-1*CHAR                                           COMCIDP    4262    A
COMCIDP            BX6    X6+X4        (X6) = ...OPI....                          COMCIDP    4263    A
COMCIDP                                                                            COMCIDP    4264    A
COMCIDP  *         ASSEMBLE NORMAL (I,J,K) ITEMS.                                  COMCIDP    4265    A
COMCIDP                                                                            COMCIDP    4266    A
COMCIDP   DAZ5     SA6    A6+B1                                                   COMCIDP    4267    A
COMCIDP            SB6    10*CHAR                                                 COMCIDP    4268    A
COMCIDP            MX6    0                                                       COMCIDP    4269    A
COMCIDP                                                                            COMCIDP    4270    A
COMCIDP   DAZ6     LX2    4                                                       COMCIDP    4271    A
COMCIDP            SB6    B6-CHAR                                                 COMCIDP    4272    A
COMCIDP            BX3    -X0*X2       (X3) = DESCRIPTOR                          COMCIDP    4273    A
COMCIDP            SA4    DAZ=I-1+X3                                              COMCIDP    4274    A
COMCIDP            ZR     X3,DAZ11     IF END OF DESCRIPTOR LIST IN SKELETON      COMCIDP    4275    A
COMCIDP                                                                            COMCIDP    4276    A
COMCIDP   DAZ7     MI     X4,DAZ8      IF THIS IS A CONDITIONAL ITEM              COMCIDP    4277    A
COMCIDP            LX4    X4,B6                                                   COMCIDP    4278    A
COMCIDP            BX6    X6+X4                                                   COMCIDP    4279    A
COMCIDP            GT     B6,B0,DAZ6   IF PACKING REG (X6) NOT FULL               COMCIDP    4280    A
COMCIDP            EQ     DAZ5                                                    COMCIDP    4281    A
COMCIDP                                                                            COMCIDP    4282    A
COMCIDP  *         PROCESS AND ASSEMBLE CONDITONAL ITEM.                           COMCIDP    4283    A
COMCIDP                                                                            COMCIDP    4284    A
COMCIDP   DAZ8     NZ     X4,DAZ9      IF *Q* 18 BIT ADDRESS                      COMCIDP    4285    A
COMCIDP                                                                            COMCIDP    4286    A
COMCIDP  *         HERE TO PROCESS CONDITIONAL *B0* ASSEMBLY.                      COMCIDP    4287    A
COMCIDP                                                                            COMCIDP    4288    A
COMCIDP            LX2    4                                                       COMCIDP    4289    A
COMCIDP            BX3    -X0*X2                                                  COMCIDP    4290    A
COMCIDP            SA4    DAZ=I-1+X3                                              COMCIDP    4291    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            SX7    X4-1R0                                                  COMCIDP     4292     A
COMCIDP            SX4    1RB                                                     COMCIDP     4293     A
COMCIDP            LX2    -4          RESTORE (X2)                                COMCIDP     4294     A
COMCIDP            NZ     X7,DAZ7     IF NOT *B0* REFERENCE                       COMCIDP     4295     A
COMCIDP                                                                            COMCIDP     4296     A
COMCIDP  *         HERE TO IGNORE *B0* REFERENCE.                                  COMCIDP     4297     A
COMCIDP                                                                            COMCIDP     4298     A
COMCIDP            LX2    2*4         SKIP OVER *B0* AND SEPARATOR                COMCIDP     4299     A
COMCIDP            SB6    B6+CHAR                                                 COMCIDP     4300     A
COMCIDP            EQ     DAZ6                                                    COMCIDP     4301     A
COMCIDP                                                                            COMCIDP     4302     A
COMCIDP  *         HERE TO PROCESS *Q* 18 BIT ADDRESS.                             COMCIDP     4303     A
COMCIDP                                                                            COMCIDP     4304     A
COMCIDP   DAZ9     SA1    DAZ=ADR+1   (X1) = 0L<Q>   (ALWAYS .LE. 6 CHARS)        COMCIDP     4305     A
COMCIDP            MX0    -CHAR                                                   COMCIDP     4306     A
COMCIDP                                                                            COMCIDP     4307     A
COMCIDP  *         ASSEMBLE *Q*.                                                   COMCIDP     4308     A
COMCIDP                                                                            COMCIDP     4309     A
COMCIDP   DAZ10    LX1    CHAR                                                    COMCIDP     4310     A
COMCIDP            BX4    -X0*X1                                                  COMCIDP     4311     A
COMCIDP            ZR     X4,DAZ11    IF FINISHED ASSEMBLING *Q*                  COMCIDP     4312     A
COMCIDP            LX4    X4,B6                                                   COMCIDP     4313     A
COMCIDP            SB6    B6-CHAR                                                 COMCIDP     4314     A
COMCIDP            BX6    X6+X4                                                   COMCIDP     4315     A
COMCIDP            GE     B6,B0,DAZ10 IF PACKING REG (X6) NOT FULL                COMCIDP     4316     A
COMCIDP            SA6    A6+B1                                                   COMCIDP     4317     A
COMCIDP            MX6    0                                                       COMCIDP     4318     A
COMCIDP            SB6    10*CHAR-1*CHAR                                          COMCIDP     4319     A
COMCIDP            EQ     DAZ10                                                   COMCIDP     4320     A
COMCIDP                                                                            COMCIDP     4321     A
COMCIDP  *         STORE FINAL WORD.                                               COMCIDP     4322     A
COMCIDP                                                                            COMCIDP     4323     A
COMCIDP   DAZ11    BX1    X6                                                      COMCIDP     4324     A
COMCIDP            RJ     =XSFN       SPACE FILL NAME                             COMCIDP     4325     A
COMCIDP            SA6    A6+1                                                    COMCIDP     4326     A
COMCIDP                                                                            COMCIDP     4327     A
COMCIDP  *         GENERATE RELATIVE ADDRESS.                                      COMCIDP     4328     A
COMCIDP                                                                            COMCIDP     4329     A
COMCIDP            SA1    DAZ=ADR                                                 COMCIDP     4330     A
COMCIDP            SA2    =10H                                                    COMCIDP     4331     A
COMCIDP            SB3    SNAPLNE+5                                               COMCIDP     4332     A
COMCIDP            SB2    A6                                                      COMCIDP     4333     A
COMCIDP            SA0    B4          SAVE (B4)                                   COMCIDP     4334     A
COMCIDP            SX0    B5+         SAVE (B5)                                   COMCIDP     4335     A
COMCIDP            GE     B2,B3,DAZ11A IF WE DONT NEED BLANK FILL                 COMCIDP     4336     A
COMCIDP            BX6    X2                                                      COMCIDP     4337     A
COMCIDP            SA6    A6+B1                                                   COMCIDP     4338     A
COMCIDP                                                                            COMCIDP     4339     A
COMCIDP   DAZ11A   SB6    30                                                      COMCIDP     4340     A
COMCIDP            MI     X1,DAZ11C   IF ADDR IS .MI. (I.E. NOT AN ADDRESS)       COMCIDP     4341     A
COMCIDP            LT     B5,B6,DAZ11C IF 15 BIT INSTRUCTION, NO RELATIVE ADDR    COMCIDP     4342     A
COMCIDP            RJ     FRA         FIND RELATIVE ADDRESS                       COMCIDP     4343     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            MI     B7,DAZ11C   IF USER DID NOT PROVIDE *FRA=*                    COMCIDP     4344     A
COMCIDP                                                                                  COMCIDP     4345     A
COMCIDP            SA1    A6          (X1) = XX........   .=BLANK(55B)                   COMCIDP     4346     A
COMCIDP  *                               X=DIFFERENT CHARS DEPENDING ON DEASSEMBLY       COMCIDP     4347     A
COMCIDP            MX2    3*CHAR                                                         COMCIDP     4348     A
COMCIDP            BX1    X2*X1       (X1) = XX.0000000   (.=BLANK(55B),0=00B)           COMCIDP     4349     A
COMCIDP            LX6    -3*CHAR                                                        COMCIDP     4350     A
COMCIDP            BX6    -X2*X6      (X6) = 000NNNNNN.   (.=BLANK(55B),0=00B)           COMCIDP     4351     A
COMCIDP            BX6    X6+X1       (X6) = XX.NNNNNN.   (.=BLANK(55B))                 COMCIDP     4352     A
COMCIDP            SA6    A1                                                             COMCIDP     4353     A
COMCIDP            SX1    3RIN                                                           COMCIDP     4354     A
COMCIDP            BX6    X7+X1       (X6) = XXXXXXXIN.   (.=BLANK(55B))                 COMCIDP     4355     A
COMCIDP            LX6    10*CHAR-3*CHAR                                                 COMCIDP     4356     A
COMCIDP            SA6    A6+B1                                                          COMCIDP     4357     A
COMCIDP                                                                                  COMCIDP     4358     A
COMCIDP  *         FINAL PROCESSING AND EXIT.                                            COMCIDP     4359     A
COMCIDP                                                                                  COMCIDP     4360     A
COMCIDP   DAZ11C   MX6    0                                                              COMCIDP     4361     A
COMCIDP            SA6    A6+B1                                                          COMCIDP     4362     A
COMCIDP            PRIDP  =XSNAPLNE                                                      COMCIDP     4363     A
COMCIDP            SB4    A0          RESTORE (B4)                                       COMCIDP     4364     A
COMCIDP            SB5    X0          RESTORE (B5)                                       COMCIDP     4365     A
COMCIDP            SB6    59-57                                                          COMCIDP     4366     A
COMCIDP            SA2    DAZ=OP                                                         COMCIDP     4367     A
COMCIDP            SA2    DAZ=PS+X2                                                      COMCIDP     4368     A
COMCIDP            LX1    X2,B6       SET TO *FORCED UPPER(.MI.)/NOT UPPER(.PL.)*        COMCIDP     4369     A
COMCIDP            EQ     EXIT.                                                          COMCIDP     4370     A
COMCIDP                                                                                  COMCIDP     4371     A
COMCIDP  *         HERE IF OUTPUTING REST OF WORD AS DATA.                               COMCIDP     4372     A
COMCIDP                                                                                  COMCIDP     4373     A
COMCIDP   DAZ12    SB5    B4          (B5) = LENGTH OF REMAINDER OF INSTR WORD           COMCIDP     4374     A
COMCIDP            SA1    A5          (X1) = INSTRUCTION WORD                            COMCIDP     4375     A
COMCIDP            RJ     VFD         VARIABLE FIELD DEFINITION                          COMCIDP     4376     A
COMCIDP            SA6    A6+B1                                                          COMCIDP     4377     A
COMCIDP            SA7    A6+B1                                                          COMCIDP     4378     A
COMCIDP            SA1    =7L   DATA                                                     COMCIDP     4379     A
COMCIDP            BX6    X1                                                             COMCIDP     4380     A
COMCIDP            SA6    A7+B1                                                          COMCIDP     4381     A
COMCIDP            SX0    B5          SAVE (B5)                                          COMCIDP     4382     A
COMCIDP            PRIDP  =XSNAPLNE                                                      COMCIDP     4383     A
COMCIDP            SB5    X0          RESTORE (B5)                                       COMCIDP     4384     A
COMCIDP            MX2    0           SET TO *THIS IS A DATA ITEM*                       COMCIDP     4385     A
COMCIDP            MX1    1           SET TO *FORCED UPPER*                              COMCIDP     4386     A
COMCIDP            EQ     EXIT.                                                          COMCIDP     4387     A
COMCIDP                                                                                  COMCIDP     4388     A
COMCIDP                                                                                  COMCIDP     4389     A
COMCIDP   DAZ=ADR  BSSZ   1           18 BIT *Q* IN BINARY                               COMCIDP     4390     A
COMCIDP            BSSZ   1           18 BIT *Q* IN -0L- FORMAT                          COMCIDP     4391     A
COMCIDP   DAZ=OPI  CON    3L   +4R                                                       COMCIDP     4392     A
COMCIDP   DAZ=OP   DATA   0           OP CODE FOR INSTRUCTION                            COMCIDP     4393     A
COMCIDP   DAZ=I    DATA   0           *I* REGISTER PORTION OF INSTRUCTION                COMCIDP     4394     A
COMCIDP   DAZ=J    DATA   0           *J* REGISTER PORTION OF INSTRUCTION                COMCIDP     4395     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP   DAZ=K    DATA   0          *K* REGISTER PORTION OF INSTRUCTION          COMCIDP   4396   A
COMCIDP   DAZ=Q    DATA   -1         FLAG TO INDICATE 18 BIT *Q*                  COMCIDP   4397   A
COMCIDP            DATA   1R+                                                     COMCIDP   4398   A
COMCIDP            DATA   1R-                                                     COMCIDP   4399   A
COMCIDP            DATA   1R*                                                     COMCIDP   4400   A
COMCIDP            DATA   1R/                                                     COMCIDP   4401   A
COMCIDP            DATA   1R,                                                     COMCIDP   4402   A
COMCIDP            DATA   1RA                                                     COMCIDP   4403   A
COMCIDP            DATA   1RB                                                     COMCIDP   4404   A
COMCIDP            DATA   -0         FLAG TO INDICATE CONDITIONAL *B0* ASSEMBLY   COMCIDP   4405   A
COMCIDP            DATA   1RX                                                     COMCIDP   4406   A
COMCIDP            DATA   1R                                                      COMCIDP   4407   A
COMCIDP   INST     SPACE  4,10                                                    COMCIDP   4408   A
COMCIDP  **        INST - MACRO TO GENERATE INSTRUCTION SKELETONS.                COMCIDP   4409   A
COMCIDP  *         GENERATES TABLE OF INSTRUCTION SKELETONS USED BY *PIG* TO      COMCIDP   4410   A
COMCIDP  *                CONVERT INSTRUCTIONS TO HUMAN-READABLE FORM FOR THE     COMCIDP   4411   A
COMCIDP  *                OBJECT CODE LISTING.                                    COMCIDP   4412   A
COMCIDP  *                                                                        COMCIDP   4413   A
COMCIDP  *         INST   (KEY),BJMP,FORCE,IJJ                                    COMCIDP   4414   A
COMCIDP  *                                                                        COMCIDP   4415   A
COMCIDP  *         *KEY*  = INSTRUCTION DESCRIPTOR.  EACH CHARACTER OF THE *KEY*  COMCIDP   4416   A
COMCIDP  *                  HAS MEANING AS FOLLOWS --                             COMCIDP   4417   A
COMCIDP  *                           I   I-PORTION OF INSTRUCTION                 COMCIDP   4418   A
COMCIDP  *                           J   J-PORTION OF INSTRUCTION                 COMCIDP   4419   A
COMCIDP  *                           K   K-PORTION OF INSTRUCTION (3 BITS ONLY)   COMCIDP   4420   A
COMCIDP  *                           Q   18-BIT *K* ADDRESS FIELD (MUST BE LAST)  COMCIDP   4421   A
COMCIDP  *                           C   OUTPUT A *B* IF THE NEXT REGISTER IS     COMCIDP   4422   A
COMCIDP  *                                       NOT A ZERO,                      COMCIDP   4423   A
COMCIDP  *                                       ELSE, SKIP THE NEXT 2 ITEMS      COMCIDP   4424   A
COMCIDP  *                       + - * / A B X , AND BLANK ALL STAND FOR          COMCIDP   4425   A
COMCIDP  *                                       THEMSELVES.                      COMCIDP   4426   A
COMCIDP  *         *BJMP* = NON-EMPTY IF THIS A *B-REGISTER* JUMP INSTRUCTION.    COMCIDP   4427   A
COMCIDP  *                          (04 THRU 07 INSTRUCTION.)                     COMCIDP   4428   A
COMCIDP  *         *FORCE*= THIS INSTRUCTION FORCES THE NEXT INSTRUCTION UPPER.   COMCIDP   4429   A
COMCIDP  *         *COPY* = COPY *J* INTO *K*.                                    COMCIDP   4430   A
COMCIDP                                                                           COMCIDP   4431   A
COMCIDP   P.COPY   =      2*CHAR                                                  COMCIDP   4432   A
COMCIDP            NOREF  D,L                                                     COMCIDP   4433   A
COMCIDP                                                                           COMCIDP   4434   A
COMCIDP                                                                           COMCIDP   4435   A
COMCIDP   INST     MACRO  KEY,BJMP,FORCE,IJJ                                      COMCIDP   4436   A
COMCIDP   A        MICRO  3,1,=KEY=                                               COMCIDP   4437   A
COMCIDP   B        MICRO  5,,=KEY=                                                COMCIDP   4438   A
COMCIDP   B        MICRO  1,,="A""B"=                                             COMCIDP   4439   A
COMCIDP   C        MICCNT B                                                       COMCIDP   4440   A
COMCIDP   D        SET    0                                                       COMCIDP   4441   A
COMCIDP   L        SET    0                                                       COMCIDP   4442   A
COMCIDP   A        MICRO  C+3,1,=KEY=                                             COMCIDP   4443   A
COMCIDP            IFC    EQ,="A"=Q=,1                                            COMCIDP   4444   A
COMCIDP   L        SET    L+4                                                     COMCIDP   4445   A
COMCIDP            IFC    NE,,BJMP,,,1                                            COMCIDP   4446   A
COMCIDP   L        SET    L+2                                                     COMCIDP   4447   A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            IFC    NE,,FORCE,,,1                                          COMCIDP    4448    A
COMCIDP   L        SET    L+1                                                    COMCIDP    4449    A
COMCIDP            VFD    3/L                                                    COMCIDP    4450    A
COMCIDP  .1        DUP    C                                                      COMCIDP    4451    A
COMCIDP   D        SET    D+1                                                    COMCIDP    4452    A
COMCIDP   A        MICRO  D,1,="B"=                                              COMCIDP    4453    A
COMCIDP   A        MICRO  2*1R"A"-1,2,/101112-D-E-F-G-H010203-L-M-N-O-P04-R-S-T-U COMCIDP    4454    A
COMCIDP  ,-V-W13-Y-Z-0-1-2-3-4-5-6-7-8-905060708-(-)-$-=1409/                    COMCIDP    4455    A
COMCIDP            VFD    4/"A"                                                  COMCIDP    4456    A
COMCIDP  .1        ENDD                                                          COMCIDP    4457    A
COMCIDP            IFLE   $,2*6-1+4,1                                            COMCIDP    4458    A
COMCIDP   8        ERR    INST - DESCRIPTOR (KEY) IS TOO LONG                    COMCIDP    4459    A
COMCIDP            POS    P.COPY+1                                               COMCIDP    4460    A
COMCIDP   A        MICRO  1,2, KEY                                               COMCIDP    4461    A
COMCIDP            VFD    1/IJJ,12/2R"A"                                         COMCIDP    4462    A
COMCIDP   INST     ENDM                                                          COMCIDP    4463    A
COMCIDP   DAZ=PS   SPACE  4,10                                                   COMCIDP    4464    A
COMCIDP  **        DAZ=PS - INSTRUCTION SKELETON TABLES FOR *DAZ*.               COMCIDP    4465    A
COMCIDP                                                                          COMCIDP    4466    A
COMCIDP                                                                          COMCIDP    4467    A
COMCIDP   DAZ=PS   INST   (PS  Q)              00    (FAKE)                      COMCIDP    4468    A
COMCIDP            INST   (RJ  Q),,1           01                                COMCIDP    4469    A
COMCIDP            INST   (JP  CI+Q),,1        02                                COMCIDP    4470    A
COMCIDP            INST   (JXI XJ,Q)           03I  (FAKE)                       COMCIDP    4471    A
COMCIDP   DAZ=EQ   INST   (EQ  CI,CJ,Q),1      04                                COMCIDP    4472    A
COMCIDP            INST   (NE  CI,CJ,Q),1      05                                COMCIDP    4473    A
COMCIDP            INST   (GE  BI,CJ,Q),1      06                                COMCIDP    4474    A
COMCIDP            INST   (LT  BI,CJ,Q),1      07                                COMCIDP    4475    A
COMCIDP                                                                          COMCIDP    4476    A
COMCIDP            INST   (BXI XJ),,,1         10                                COMCIDP    4477    A
COMCIDP            INST   (BXI XJ*XK)          11                                COMCIDP    4478    A
COMCIDP            INST   (BXI XJ+XK)          12                                COMCIDP    4479    A
COMCIDP            INST   (BXI XJ-XK)          13                                COMCIDP    4480    A
COMCIDP            INST   (BXI -XJ),,,1        14                                COMCIDP    4481    A
COMCIDP            INST   (BXI -XK*XJ)         15                                COMCIDP    4482    A
COMCIDP            INST   (BXI -XK+XJ)         16                                COMCIDP    4483    A
COMCIDP            INST   (BXI -XJ-XK)         17                                COMCIDP    4484    A
COMCIDP                                                                          COMCIDP    4485    A
COMCIDP            INST   (LXI JKB)            20                                COMCIDP    4486    A
COMCIDP            INST   (AXI JKB)            21                                COMCIDP    4487    A
COMCIDP            INST   (LXI CJ,XK)          22                                COMCIDP    4488    A
COMCIDP            INST   (AXI CJ,XK)          23                                COMCIDP    4489    A
COMCIDP            INST   (NXI CJ,XK)          24                                COMCIDP    4490    A
COMCIDP            INST   (ZXI CJ,XK)          25                                COMCIDP    4491    A
COMCIDP            INST   (UXI CJ,XK)          26                                COMCIDP    4492    A
COMCIDP            INST   (PXI CJ,XK)          27                                COMCIDP    4493    A
COMCIDP                                                                          COMCIDP    4494    A
COMCIDP            INST   (FXI XJ+XK)          30                                COMCIDP    4495    A
COMCIDP            INST   (FXI XJ-XK)          31                                COMCIDP    4496    A
COMCIDP            INST   (DXI XJ+XK)          32                                COMCIDP    4497    A
COMCIDP            INST   (DXI XJ-XK)          33                                COMCIDP    4498    A
COMCIDP            INST   (RXI XJ+XK)          34                                COMCIDP    4499    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            INST   (RXI XJ-XK)          35                                  COMCIDP     4500     A
COMCIDP            INST   (IXI XJ+XK)          36                                  COMCIDP     4501     A
COMCIDP            INST   (IXI XJ-XK)          37                                  COMCIDP     4502     A
COMCIDP                                                                            COMCIDP     4503     A
COMCIDP            INST   (FXI XJ*XK)          40                                  COMCIDP     4504     A
COMCIDP            INST   (RXI XJ*XK)          41                                  COMCIDP     4505     A
COMCIDP            INST   (DXI XJ*XK)          42                                  COMCIDP     4506     A
COMCIDP            INST   (MXI JKB)            43                                  COMCIDP     4507     A
COMCIDP            INST   (FXI XJ/XK)          44                                  COMCIDP     4508     A
COMCIDP            INST   (RXI XJ/XK)          45                                  COMCIDP     4509     A
COMCIDP            INST   (NO  IJKB)           46                                  COMCIDP     4510     A
COMCIDP            INST   (CXI XK),,,1         47                                  COMCIDP     4511     A
COMCIDP                                                                            COMCIDP     4512     A
COMCIDP            INST   (SAI AJ+Q)           50                                  COMCIDP     4513     A
COMCIDP            INST   (SAI CJ+Q)           51                                  COMCIDP     4514     A
COMCIDP            INST   (SAI XJ+Q)           52                                  COMCIDP     4515     A
COMCIDP            INST   (SAI CK+XJ)          53                                  COMCIDP     4516     A
COMCIDP            INST   (SAI CK+AJ)          54                                  COMCIDP     4517     A
COMCIDP            INST   (SAI AJ-BK)          55                                  COMCIDP     4518     A
COMCIDP            INST   (SAI CK+BJ)          56                                  COMCIDP     4519     A
COMCIDP            INST   (SAI -BK+BJ)         57                                  COMCIDP     4520     A
COMCIDP                                                                            COMCIDP     4521     A
COMCIDP            INST   (SBI AJ+Q)           60                                  COMCIDP     4522     A
COMCIDP            INST   (SBI CJ+Q)           61                                  COMCIDP     4523     A
COMCIDP            INST   (SBI XJ+Q)           62                                  COMCIDP     4524     A
COMCIDP            INST   (SBI CK+XJ)          63                                  COMCIDP     4525     A
COMCIDP            INST   (SBI CK+AJ)          64                                  COMCIDP     4526     A
COMCIDP            INST   (SBI AJ-BK)          65                                  COMCIDP     4527     A
COMCIDP            INST   (SBI CK+BJ)          66                                  COMCIDP     4528     A
COMCIDP            INST   (SBI -BK+BJ)         67                                  COMCIDP     4529     A
COMCIDP                                                                            COMCIDP     4530     A
COMCIDP            INST   (SXI AJ+Q)           70                                  COMCIDP     4531     A
COMCIDP            INST   (SXI CJ+Q)           71                                  COMCIDP     4532     A
COMCIDP            INST   (SXI XJ+Q)           72                                  COMCIDP     4533     A
COMCIDP            INST   (SXI CK+XJ)          73                                  COMCIDP     4534     A
COMCIDP            INST   (SXI CK+AJ)          74                                  COMCIDP     4535     A
COMCIDP            INST   (SXI AJ-BK)          75                                  COMCIDP     4536     A
COMCIDP            INST   (SXI CK+BJ)          76                                  COMCIDP     4537     A
COMCIDP            INST   (SXI -BK+BJ)         77                                  COMCIDP     4538     A
COMCIDP                                                                            COMCIDP     4539     A
COMCIDP   DAZ=XJP  INST   (ZR  XJ,Q)           030                                 COMCIDP     4540     A
COMCIDP            INST   (NZ  XJ,Q)           031                                 COMCIDP     4541     A
COMCIDP            INST   (PL  XJ,Q)           032                                 COMCIDP     4542     A
COMCIDP            INST   (MI  XJ,Q)           033                                 COMCIDP     4543     A
COMCIDP            INST   (IR  XJ,Q)           034                                 COMCIDP     4544     A
COMCIDP            INST   (OR  XJ,Q)           035                                 COMCIDP     4545     A
COMCIDP            INST   (DF  XJ,Q)           036                                 COMCIDP     4546     A
COMCIDP            INST   (ID  XJ,Q)           037                                 COMCIDP     4547     A
COMCIDP   DCM      SPACE  4,10                                                     COMCIDP     4548     A
COMCIDP   .IDPOS   ENDIF                                                           CCGA018       19     A
COMCIDP  **        DCM - DUMP CENTRAL MEMORY.                                      COMCIDP     4549     A
COMCIDP  *                                                                         COMCIDP     4550     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *                                                                  COMCIDP    4551    A
COMCIDP  *                CONVERTS CM WORDS TO OCTAL DPC AND DUMPS THEM IN THE  COMCIDP    4552    A
COMCIDP  *         FOLLOWING FORMAT--                                       COMCIDP    4553    A
COMCIDP  *     COL 1         1         1         1         1               COMCIDP    4554    A
COMCIDP  *         .NNNNNN.+.CCCCCCCCCCCCCCCCCCCC.........ETC               COMCIDP    4555    A
COMCIDP  *                   ******************************               COMCIDP    4556    A
COMCIDP  *         N = ADDR OF 1ST WORD DUMPED ON LINE                      COMCIDP    4557    A
COMCIDP  *         + = + IF *NNNNNN* IS DECK RELATIVE, ELSE                 COMCIDP    4558    A
COMCIDP  *           = BLANK(55B) IF *NNNNNN* IS ABSOLUTE ADDR.             COMCIDP    4559    A
COMCIDP  *         C = CONVERTED CM WORD                                    COMCIDP    4560    A
COMCIDP  *         . = BLANK(55B)                                           COMCIDP    4561    A
COMCIDP  *         THE FORMAT OF COLS 11 THRU 40 (INDICATED BY *) IS REPEATED FOR  COMCIDP    4562    A
COMCIDP  *           HOWEVER MANY CM WORDS ARE TO BE DUMPED ON A LINE.      COMCIDP    4563    A
COMCIDP  *                                                                  COMCIDP    4564    A
COMCIDP  *                (IDPFLG) = 42/OTHER, 18/NR OF CM WORDS TO BE DUMPED ON  COMCIDP    4565    A
COMCIDP  *                             A SINGLE LINE                        COMCIDP    4566    A
COMCIDP  *                                                                  COMCIDP    4567    A
COMCIDP  *         ENTRY  (X2) = 18/0, 21/LEV OF IND ADDR ,21/FWA           COMCIDP    4568    A
COMCIDP  *                (X3) = 18/0, 21/LEV OF IND ADDR ,21/LWA           COMCIDP    4569    A
COMCIDP  *                        (X3) IS OPTIONAL                          COMCIDP    4570    A
COMCIDP  *                (X4) = 18/0,21/LEV OF IND ADDR ,21/LEN            COMCIDP    4571    A
COMCIDP  *                        NOTE -- IF LEN = 0, DMP IS FROM FWA TO LWA,  COMCIDP    4572    A
COMCIDP  *                                     ELSE DMP IS FROM FWA TO FWA+LEN  COMCIDP    4573    A
COMCIDP  *                                                                  COMCIDP    4574    A
COMCIDP  *         EXIT   (B2) = .MI. IF FWA,LWA, OR LEN BAD, ELSE .PL.     COMCIDP    4575    A
COMCIDP  *                                                                  COMCIDP    4576    A
COMCIDP  *         USES   ALL BUT A0  (INCLUDES ALL CALLS)                  COMCIDP    4577    A
COMCIDP  *                                                                  COMCIDP    4578    A
COMCIDP  *         CALLS  CAD,FLL,PRIDP,WOD                                 COMCIDP    4579    A
COMCIDP                                                                     COMCIDP    4580    A
COMCIDP                                                                     COMCIDP    4581    A
COMCIDP   DCM      SUBR                ** ENTRY/EXIT **                     COMCIDP    4582    A
COMCIDP            RJ     FLL          CHECK FWA,LWA, AND LENGTH            COMCIDP    4583    A
COMCIDP            SA1    =10H                                              COMCIDP    4584    A
COMCIDP            MI     B2,EXIT.     IF FWA,LWA, OR LEN IS BAD            COMCIDP    4585    A
COMCIDP            BX6    X1                                                COMCIDP    4586    A
COMCIDP            SB6    B0           PRESET TO *READY FOR NEW SNAP LINE*  COMCIDP    4587    A
COMCIDP            SA6    SNAPLNE      PRESTORE (A6) = ADDR OF LAST WORD STORED  COMCIDP    4588    A
COMCIDP  *                               INTO SNAP LINE.  THIS WILL FORCE A BLANK  COMCIDP    4589    A
COMCIDP  *                               LINE TO BE OUTPUT AS 1ST LINE OF DUMP.  COMCIDP    4590    A
COMCIDP                                                                     COMCIDP    4591    A
COMCIDP  *         ** MAIN LOOP NODE **                                     COMCIDP    4592    A
COMCIDP                                                                     COMCIDP    4593    A
COMCIDP   DCM2     SA5    AP=FWA       (X5) = ADDR OF NEXT WORD TO DUMP     COMCIDP    4594    A
COMCIDP            SA2    A5+B1        (X2) = LWA OF DUMP                   COMCIDP    4595    A
COMCIDP            IX3    X2-X5                                             COMCIDP    4596    A
COMCIDP            SX7    X5+B1                                             COMCIDP    4597    A
COMCIDP            MI     X3,DCM6      IF FINISHED                          COMCIDP    4598    A
COMCIDP            SA7    A5                                                COMCIDP    4599    A
COMCIDP            GT     B6,B0,DCM3   IF MORE WORDS TO DUMP ON THIS LINE   COMCIDP    4600    A
COMCIDP                                                                     COMCIDP    4601    A
COMCIDP  *         HERE IF READY TO OUTPUT OLD LINE AND START NEW LINE BY   COMCIDP    4602    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *          SETTING UP ADDR OF 1ST WORD ON LINE.                    COMCIDP    4603    A
COMCIDP                                                                     COMCIDP    4604    A
COMCIDP            MX6    0                                                 COMCIDP    4605    A
COMCIDP            SA6    A6+B1       MARK EOL                              COMCIDP    4606    A
COMCIDP            PRIDP  SNAPLNE                                           COMCIDP    4607    A
COMCIDP                                                                     COMCIDP    4608    A
COMCIDP            SX1    X5                                                COMCIDP    4609    A
COMCIDP            RJ     CAD         CONVERT ADDR TO DPC                   COMCIDP    4610    A
COMCIDP            SA1    IDPFLG                                            COMCIDP    4611    A
COMCIDP            SB6    X1          (B6) = NR OF WORDS TO DUMP ON A LINE  COMCIDP    4612    A
COMCIDP            SA6    SNAPLNE                                           COMCIDP    4613    A
COMCIDP            LE     B6,B0,*+4S15 IF BAD WORDS PER LINE COUNT...       COMCIDP    4614    A
COMCIDP                                                                     COMCIDP    4615    A
COMCIDP  *         HERE TO DUMP A SINGLE WORD.                              COMCIDP    4616    A
COMCIDP                                                                     COMCIDP    4617    A
COMCIDP   DCM3     SA1    X5          (X1) = NEXT WORD TO BE DUMPED         COMCIDP    4618    A
COMCIDP            SA2    =10H                                              COMCIDP    4619    A
COMCIDP            NZ     X1,DCM4     IF WORD TO DUMP IS NON-ZERO           COMCIDP    4620    A
COMCIDP            SX3    2R 0&2R                                           COMCIDP    4621    A
COMCIDP            SX4    2R-0&2R                                           COMCIDP    4622    A
COMCIDP            BX3    -X1*X3                                            COMCIDP    4623    A
COMCIDP            BX4    -X1+X4                                            COMCIDP    4624    A
COMCIDP            IX5    X3+X4                                             COMCIDP    4625    A
COMCIDP            LX6    X2                                                COMCIDP    4626    A
COMCIDP            BX7    X2-X5       (X7) = .........0   (.=55B)  IF (X1) = +0  COMCIDP    4627    A
COMCIDP  *                                 = ........-0   (.=55B)  IF (X1) = -0  COMCIDP    4628    A
COMCIDP            EQ     DCM5                                              COMCIDP    4629    A
COMCIDP                                                                     COMCIDP    4630    A
COMCIDP   DCM4     RJ     =XWOD       CONVERT A FULL BINARY WORD TO OCTAL DPC  COMCIDP    4631    A
COMCIDP                                                                     COMCIDP    4632    A
COMCIDP   DCM5     SA6    A6+B1                                             COMCIDP    4633    A
COMCIDP            SA7    A6+B1                                             COMCIDP    4634    A
COMCIDP            SA1    =10H                                              COMCIDP    4635    A
COMCIDP            SB6    B6-B1                                             COMCIDP    4636    A
COMCIDP            BX6    X1                                                COMCIDP    4637    A
COMCIDP            SA6    A7+B1                                             COMCIDP    4638    A
COMCIDP            EQ     DCM2        MAIN LOOP...                          COMCIDP    4639    A
COMCIDP                                                                     COMCIDP    4640    A
COMCIDP  *         HERE IF FINISHED -- NEED TO OUTPUT FINAL WORD.           COMCIDP    4641    A
COMCIDP                                                                     COMCIDP    4642    A
COMCIDP   DCM6     MX6    0                                                 COMCIDP    4643    A
COMCIDP            SA6    A6+B1       MARK EOL                              COMCIDP    4644    A
COMCIDP            PRIDP  SNAPLNE                                           COMCIDP    4645    A
COMCIDP            EQ     EXIT.                                             COMCIDP    4646    A
COMCIDP   DOD      SPACE  4,10                                              COMCIDP    4647    A
COMCIDP   .IDPOS   IFNE   .OS,2                                             CCGA018      20    A
COMCIDP  **        DOD - DUMP CENTRAL MEMORY -- OCTAL AND DPC.              COMCIDP    4648    A
COMCIDP  *                                                                  COMCIDP    4649    A
COMCIDP  *                                                                  COMCIDP    4650    A
COMCIDP  *                CONVERTS CM WORDS TO OCTAL DPC AND ALSO DUMPS THE COMCIDP    4651    A
COMCIDP  *         UNCONVERTED DPC EQUIVALENT IN THE FOLLOWING FORMAT--     COMCIDP    4652    A
COMCIDP  *     COL 1         1         1         1         1         1      COMCIDP    4653    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *         .NNNNNN.+.CCCCCCCCCCCCCCCCCCCC..........UUUUUUUUUU                COMCIDP    4654    A
COMCIDP  *         N = ADDR OF WORD DUMPED                                          COMCIDP    4655    A
COMCIDP  *         + = + IF *NNNNNN* IS DECK RELATIVE, ELSE                         COMCIDP    4656    A
COMCIDP  *           = BLANK(55B) IF *NNNNNN* IS ABSOLUTE ADDR.                     COMCIDP    4657    A
COMCIDP  *         C = CONVERTED CM WORD                                            COMCIDP    4658    A
COMCIDP  *         U = UNCONVERTED CM WORD. NOTE--TRAILING COLONS ARE TREATED AS    COMCIDP    4659    A
COMCIDP  *               AN EOL AND THEREFORE WILL NOT APPEAR.                      COMCIDP    4660    A
COMCIDP  *         . = BLANK(55B)                                                   COMCIDP    4661    A
COMCIDP  *                                                                          COMCIDP    4662    A
COMCIDP  *         ENTRY  (X2) = 18/0,21/LEV OF IND ADDR,21/FWA                     COMCIDP    4663    A
COMCIDP  *                (X3) = 18/0,21/LEV OF IND ADDR,21/LWA                     COMCIDP    4664    A
COMCIDP  *                       NOTE-- (X3) IS OPTIONAL                            COMCIDP    4665    A
COMCIDP  *                (X4) = 18/0,21/LEV OF IND ADDR,21/LEN                     COMCIDP    4666    A
COMCIDP  *                       NOTE-- IF LEN = 0, DMP IS FROM FWA TO LWA,         COMCIDP    4667    A
COMCIDP  *                       ELSE DMP IS FROM FWA TO FWA+LEN-1                  COMCIDP    4668    A
COMCIDP  *                                                                          COMCIDP    4669    A
COMCIDP  *         EXIT   (B2) = .MI. IF FWA,LWA, OR LEN IS BAD, ELSE .PL.          COMCIDP    4670    A
COMCIDP  *                                                                          COMCIDP    4671    A
COMCIDP  *         USES   ALL                        (INCLUDES ALL CALLS)           COMCIDP    4672    A
COMCIDP  *                                                                          COMCIDP    4673    A
COMCIDP  *         CALLS  CAD,FLL,PRIDP,WOD,ZTB                                     COMCIDP    4674    A
COMCIDP                                                                             COMCIDP    4675    A
COMCIDP                                                                             COMCIDP    4676    A
COMCIDP   DOD      SUBR                ** ENTRY/EXIT **                             COMCIDP    4677    A
COMCIDP            RJ     FLL          CHECK FWA,LWA, AND LENGTH                    COMCIDP    4678    A
COMCIDP            MI     B2,EXIT.     IF FWA,LWA, OR LEN IS BAD                    COMCIDP    4679    A
COMCIDP                                                                             COMCIDP    4680    A
COMCIDP  *         HERE TO BUILD SNAP OUTPUT LINE.                                  COMCIDP    4681    A
COMCIDP                                                                             COMCIDP    4682    A
COMCIDP   DOD2     SA1    AP=FWA       (X1) = ADDR OF NEXT WORD TO DUMP             COMCIDP    4683    A
COMCIDP            SA2    A1+B1        (X2) = LWA TO DUMP                           COMCIDP    4684    A
COMCIDP            IX3    X2-X1                                                     COMCIDP    4685    A
COMCIDP            SX6    X1+B1                                                     COMCIDP    4686    A
COMCIDP            SB2    B0           SET TO *NO ERROR OCCURRED*                   COMCIDP    4687    A
COMCIDP            MI     X3,EXIT.     IF FINISHED                                  COMCIDP    4688    A
COMCIDP            SA6    A1                                                        COMCIDP    4689    A
COMCIDP            SA0    X1                                                        COMCIDP    4690    A
COMCIDP            RJ     CAD          CONVERT ADDR TO DPC                          COMCIDP    4691    A
COMCIDP            SA6    SNAPLNE                                                   COMCIDP    4692    A
COMCIDP            SA1    A0           (X1) = NEXT WORD TO BE DUMPED                COMCIDP    4693    A
COMCIDP            RJ     =XWOD        CONVERT A FULL BINARY WORD TO OCTAL DPC      COMCIDP    4694    A
COMCIDP            SA6    A6+B1                                                     COMCIDP    4695    A
COMCIDP            SA7    A6+B1                                                     COMCIDP    4696    A
COMCIDP            SA1    =10H                                                      COMCIDP    4697    A
COMCIDP            BX6    X1                                                        COMCIDP    4698    A
COMCIDP            SA6    A7+B1                                                     COMCIDP    4699    A
COMCIDP                                                                             COMCIDP    4700    A
COMCIDP            SA1    A0+          (X1) = WORD TO BE DUMPED                     COMCIDP    4701    A
COMCIDP            RJ     =XZTB        CONVERT ZEROES TO BLANKS                     COMCIDP    4702    A
COMCIDP            SA6    A6+1                                                      COMCIDP    4703    A
COMCIDP                                                                             COMCIDP    4704    A
COMCIDP            BX6    X6-X6                                                     COMCIDP    4705    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            SA6    A6+B1       MARK EOL                                          COMCIDP     4706     A
COMCIDP            PRIDP  SNAPLNE                                                       COMCIDP     4707     A
COMCIDP            EQ     DOD2                                                          COMCIDP     4708     A
COMCIDP   .IDPOS   ENDIF                                                                CCGA018       21     A
COMCIDP   DSR      SPACE  4,10                                                          COMCIDP     4709     A
COMCIDP  **        DSR - DUMP SELECTED REGISTERS.                                       COMCIDP     4710     A
COMCIDP  *                                                                              COMCIDP     4711     A
COMCIDP  *                                                                              COMCIDP     4712     A
COMCIDP  *                PERFORMS A SELECTED REGISTER DUMP, BASED ON A PARAMETER       COMCIDP     4713     A
COMCIDP  *         LIST SET UP BY *SNAP* AND/OR *REG* MACROS.                           COMCIDP     4714     A
COMCIDP  *                                                                              COMCIDP     4715     A
COMCIDP  *                0TR = REGISTER DESIGNATOR                                     COMCIDP     4716     A
COMCIDP  *                           T = REGISTER TYPE                                  COMCIDP     4717     A
COMCIDP  *                             = 0 FOR -B- REGISTER                             COMCIDP     4718     A
COMCIDP  *                             = 1 FOR -A- REGISTER                             COMCIDP     4719     A
COMCIDP  *                             = 2 FOR -X- REGISTER                             COMCIDP     4720     A
COMCIDP  *                           R = REGISTER NR (0-7)                              COMCIDP     4721     A
COMCIDP  *                             E.G. A3 WOULD BE 013B                            COMCIDP     4722     A
COMCIDP  *                IDP=SVB = FWA OF REGISTER SAVE AREA                           COMCIDP     4723     A
COMCIDP  *                                                                              COMCIDP     4724     A
COMCIDP  *         PARAMETER LIST FORMAT--                                              COMCIDP     4725     A
COMCIDP  *                                                                              COMCIDP     4726     A
COMCIDP  *         VFD    60/IDP=SVB+0TR                                                COMCIDP     4727     A
COMCIDP  *         VFD    60/IDP=SVB+0TR                                                COMCIDP     4728     A
COMCIDP  *                 .                                                            COMCIDP     4729     A
COMCIDP  *                 .                                                            COMCIDP     4730     A
COMCIDP  *                60/0        END OF PARAMETER LIST                             COMCIDP     4731     A
COMCIDP  *                                                                              COMCIDP     4732     A
COMCIDP  *         ENTRY  (A1,X1) = ADDR + CNTS OF 1ST WORD OF PARAMETER LIST           COMCIDP     4733     A
COMCIDP  *                                                                              COMCIDP     4734     A
COMCIDP  *         EXIT   NONE                                                          COMCIDP     4735     A
COMCIDP  *                                                                              COMCIDP     4736     A
COMCIDP  *         USES   ALL                                                           COMCIDP     4737     A
COMCIDP  *                                                                              COMCIDP     4738     A
COMCIDP  *         CALLS  DAB,DUX                                                       COMCIDP     4739     A
COMCIDP                                                                                 COMCIDP     4740     A
COMCIDP                                                                                 COMCIDP     4741     A
COMCIDP   DSR      SUBR               ** ENTRY/EXIT **                                  COMCIDP     4742     A
COMCIDP                                                                                 COMCIDP     4743     A
COMCIDP   DSR2     ZR     X1,EXIT.    IF FINISHED DUMPING SELECTED REGISTERS            COMCIDP     4744     A
COMCIDP            SB2    IDP=SVB                                                       COMCIDP     4745     A
COMCIDP            SB3    X1                                                            COMCIDP     4746     A
COMCIDP            SX6    A1+                                                           COMCIDP     4747     A
COMCIDP            SB4    B3-B2                                                         COMCIDP     4748     A
COMCIDP            SB5    20B                                                           COMCIDP     4749     A
COMCIDP            SA6    DSRA        SAVE (A1) = CURRENT POSITION IN PARM LIST         COMCIDP     4750     A
COMCIDP            GE     B4,B5,DSR3  IF -X- REGISTER DUMP                              COMCIDP     4751     A
COMCIDP            RJ     DAB         DUMP -A- OR -B- REGISTER                          COMCIDP     4752     A
COMCIDP            EQ     DSR4                                                          COMCIDP     4753     A
COMCIDP                                                                                 COMCIDP     4754     A
COMCIDP   DSR3     RJ     DUX         DUMP -X- REGISTER                                 COMCIDP     4755     A
COMCIDP                                                                                 COMCIDP     4756     A
```

```
         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP   DSR4     SA1    DSRA                                                    COMCIDP     4757     A
COMCIDP            SA1    X1+1                                                    COMCIDP     4758     A
COMCIDP            EQ     DSR2                                                    COMCIDP     4759     A
COMCIDP                                                                           COMCIDP     4760     A
COMCIDP                                                                           COMCIDP     4761     A
COMCIDP   DSRA     BSSZ   1           SAVED (A1) = ADDR OF NEXT REGISTER PARAME-  COMCIDP     4762     A
COMCIDP  *                              TER WORD                                  COMCIDP     4763     A
COMCIDP   DUX      SPACE  4,10                                                    COMCIDP     4764     A
COMCIDP  **        DUX - DUMP AN -X- REGISTER.                                    COMCIDP     4765     A
COMCIDP  *                                                                        COMCIDP     4766     A
COMCIDP  *                                                                        COMCIDP     4767     A
COMCIDP  *         ENTRY  (B4) = 0TR, WHERE                                       COMCIDP     4768     A
COMCIDP  *                           T IS REGISTER TYPE (2=X REGISTER)            COMCIDP     4769     A
COMCIDP  *                           R IS REGISTER NR (0-7)                       COMCIDP     4770     A
COMCIDP  *                              E.G. X1 WOULD BE 021B                     COMCIDP     4771     A
COMCIDP  *                                                                        COMCIDP     4772     A
COMCIDP  *         EXIT   (B4) IS UNCHANGED                                       COMCIDP     4773     A
COMCIDP  *                                                                        COMCIDP     4774     A
COMCIDP  *         USES   ALL BUT B4                                              COMCIDP     4775     A
COMCIDP  *                                                                        COMCIDP     4776     A
COMCIDP  *         CALLS  PRIDP,WOD,ZTB                                           COMCIDP     4777     A
COMCIDP                                                                           COMCIDP     4778     A
COMCIDP                                                                           COMCIDP     4779     A
COMCIDP   DUX      SUBR               ** ENTRY/EXIT **                            COMCIDP     4780     A
COMCIDP            SA2    DUXA                                                    COMCIDP     4781     A
COMCIDP            SX7    B4-20B      (X7) = X REGISTER NR                        COMCIDP     4782     A
COMCIDP            SA1    IDP=SVX+X7   (X1) = SAVED -X- REGISTER                  COMCIDP     4783     A
COMCIDP            LX7    2*CHAR-1*CHAR                                           COMCIDP     4784     A
COMCIDP            IX6    X2+X7       (X6) = 10H       XN  , WHERE N=0 THRU 7     COMCIDP     4785     A
COMCIDP            SA6    SNAPLNE                                                 COMCIDP     4786     A
COMCIDP            RJ     =XWOD       CONVERT ONE FULL BINARY WORD TO OCTAL DPC   COMCIDP     4787     A
COMCIDP            SA6    A6+B1                                                   COMCIDP     4788     A
COMCIDP            SA7    A6+B1                                                   COMCIDP     4789     A
COMCIDP            SA1    =10H                                                    COMCIDP     4790     A
COMCIDP            BX6    X1                                                      COMCIDP     4791     A
COMCIDP            SA6    A7+B1                                                   COMCIDP     4792     A
COMCIDP                                                                           COMCIDP     4793     A
COMCIDP            SA1    IDP=SVB+B4  (X1) = SAVED CONTENTS OF X-N-               COMCIDP     4794     A
COMCIDP            RJ     =XZTB       CONVERT ZEROES TO BLANKS                    COMCIDP     4795     A
COMCIDP            SA6    A6+1                                                    COMCIDP     4796     A
COMCIDP                                                                           COMCIDP     4797     A
COMCIDP            BX6    X6-X6                                                   COMCIDP     4798     A
COMCIDP            SA6    A6+B1       MARK EOL                                    COMCIDP     4799     A
COMCIDP            PRIDP  SNAPLNE                                                 COMCIDP     4800     A
COMCIDP            EQ     EXIT.                                                   COMCIDP     4801     A
COMCIDP                                                                           COMCIDP     4802     A
COMCIDP                                                                           COMCIDP     4803     A
COMCIDP   DUXA     DATA   10H       X0                                            COMCIDP     4804     A
COMCIDP   FAA      SPACE  4,10                                                    COMCIDP     4805     A
COMCIDP   .IDPOS   IFNE   .OS,2                                                   CCGA018       22     A
COMCIDP  **        FAA - FIND ABSOLUTE ADDRESS.                                   COMCIDP     4806     A
COMCIDP  *                                                                        COMCIDP     4807     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *                                                                  COMCIDP    4808    A
COMCIDP  *         THIS ROUTINE TRIES TO ASSOCIATE A BINARY ADDR/NUMBER     COMCIDP    4809    A
COMCIDP  *         WITH A GIVEN NAME IN -L- FORMAT.  THIS IS USED BY IDP    COMCIDP    4810    A
COMCIDP  *         WHEN THE PROGRAMMER ENTERS, FOR EXAMPLE, A DECK NAME.    COMCIDP    4811    A
COMCIDP  *                                                                  COMCIDP    4812    A
COMCIDP  *         E.G. --                                                  COMCIDP    4813    A
COMCIDP  *                                                                  COMCIDP    4814    A
COMCIDP  *             CODE DECK+1123                                       COMCIDP    4815    A
COMCIDP  *                                                                  COMCIDP    4816    A
COMCIDP  *         *FAA* TRIES TO ASSOCIATE A BINARY VALUE WITH *DECK*      COMCIDP    4817    A
COMCIDP  *         IN THE FOLLOWING MANNER --                               COMCIDP    4818    A
COMCIDP  *                                                                  COMCIDP    4819    A
COMCIDP  *           1. IF THE USER PROVIDED AN *FAA=*, THEN THIS *FAA*     COMCIDP    4820    A
COMCIDP  *              ASSUMES THAT THE USER WISHES TO PERFORM HIS OWN     COMCIDP    4821    A
COMCIDP  *              ADDR TRANSLATION.  THEREFORE, *FAA* MERELY CALLS *FAA=*.  COMCIDP  4822  A
COMCIDP  *                                                                  COMCIDP    4823    A
COMCIDP  *           2. IF THE HOST IS USING *CCOMRPV* (COMMON REPRIEVE     COMCIDP    4824    A
COMCIDP  *              PROCESSOR), THEN *FAA* WILL ATTEMPT TO QUERY THE LINKED  COMCIDP  4825  A
COMCIDP  *              TABLES USED BY *CCOMRPV* THAT CONTAIN ALL THE ROUTINE  COMCIDP  4826  A
COMCIDP  *              NAMES AND FWAS.  SEE *CCOMRPV* FOR DETAILS.         COMCIDP    4827    A
COMCIDP  *                                                                  COMCIDP    4828    A
COMCIDP  *         ENTRY  (X0) = MASK TO EXTRACT NAMES, I.E. MX0 7*CHAR.    COMCIDP    4829    A
COMCIDP  *                (X1) = NAME TO SEARCH FOR, IN -L- FMT.            COMCIDP    4830    A
COMCIDP  *                                                                  COMCIDP    4831    A
COMCIDP  *         EXIT   (X6) = .PL. IF ABS ADDR IDP IS TO USE FOR *NAME*. COMCIDP    4832    A
COMCIDP  *                     = .MI. IF NO ADDR KNOWN FOR *NAME* (I.E. NO FIND).  COMCIDP  4833  A
COMCIDP  *                                                                  COMCIDP    4834    A
COMCIDP  *         USES   CANNOT DESTROY A0,A5,X5                           COMCIDP    4835    A
COMCIDP  *                                                                  COMCIDP    4836    A
COMCIDP  *         CALLS  FAA=(IF AVAIL)                                    COMCIDP    4837    A
COMCIDP                                                                     COMCIDP    4838    A
COMCIDP                                                                     COMCIDP    4839    A
COMCIDP   FAA      SUBR                ** ENTRY/EXIT **                     COMCIDP    4840    A
COMCIDP            SX6    =YFAA=                                            COMCIDP    4841    A
COMCIDP            SB2    =YRPV=RNA                                         COMCIDP    4842    A
COMCIDP            MI     X6,FAA2     IF USER DID NOT PROVIDE *FAA=*        COMCIDP    4843    A
COMCIDP            RJ     =YFAA=      FIND ABSOLUTE ADDRESS                 COMCIDP    4844    A
COMCIDP            EQ     EXIT.                                             COMCIDP    4845    A
COMCIDP                                                                     COMCIDP    4846    A
COMCIDP   FAA2     MI     B2,EXIT.    IF USER IS NOT USING *CCOMRPV*        COMCIDP    4847    A
COMCIDP            SA2    B2          (X2) = FWA OF 1ST RNA TABLE           COMCIDP    4848    A
COMCIDP            SA2    X2          (A2,X2) = A+C OF 1ST RPV NAME ADDR ENTRY  COMCIDP  4849  A
COMCIDP                                                                     COMCIDP    4850    A
COMCIDP   FAA3     ZR     X2,FAA4     IF END-OF-TABLE ENCOUNTERED           COMCIDP    4851    A
COMCIDP            BX7    X0*X2       (X7) = NAME IN *RPV* TABLE, -L- FMT   COMCIDP    4852    A
COMCIDP            IX3    X7-X1                                             COMCIDP    4853    A
COMCIDP            ZR     X3,FAA5     IF A HIT                              COMCIDP    4854    A
COMCIDP            SA2    A2+B1                                             COMCIDP    4855    A
COMCIDP            EQ     FAA3                                              COMCIDP    4856    A
COMCIDP                                                                     COMCIDP    4857    A
COMCIDP   FAA4     SA3    A2+B1       (X3) = LINK TO NEXT TABLE, ELSE .ZR.  COMCIDP    4858    A
COMCIDP            ZR     X3,EXIT.    IF END OF ALL TABLES ENCOUNTERED      COMCIDP    4859    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            SA2    X3          (A2,X2) = A+C OF 1ST ENTRY IN NEXT TABLE          COMCIDP    4860    A
COMCIDP            EQ     FAA3                                                          COMCIDP    4861    A
COMCIDP                                                                                  COMCIDP    4862    A
COMCIDP   FAA5     SX6    X2          (X6) = ABS ADDR FOR THIS NAME                     COMCIDP    4863    A
COMCIDP            EQ     EXIT.                                                         COMCIDP    4864    A
COMCIDP   .IDPOS   ENDIF                                                                CCGA018      23    A
COMCIDP   FAB      SPACE  4,10                                                          COMCIDP    4865    A
COMCIDP  **        FAB - FORMAT AN -A- OR -B- REGISTER.                                 COMCIDP    4866    A
COMCIDP  *                                                                              COMCIDP    4867    A
COMCIDP  *                                                                              COMCIDP    4868    A
COMCIDP  *                 THIS ROUTINE WILL FORMAT AN -A- OR -B- REGISTER              COMCIDP    4869    A
COMCIDP  *         SUITABLE FOR PRINTING. FORMAT--                                      COMCIDP    4870    A
COMCIDP  *                                                                              COMCIDP    4871    A
COMCIDP  *    (B6)+0         0         0         0         0         0                 COMCIDP    4872    A
COMCIDP  *         .......TR..NNNNNN.....C(TR).=.NNNNNNNNNNNNNNNNNNNN                    COMCIDP    4873    A
COMCIDP  *                                                                              COMCIDP    4874    A
COMCIDP  *         T=REGISTER TYPE (B,A),R=REGISTER NR (0-7),.=BLANK(55B)               COMCIDP    4875    A
COMCIDP  *                                                                              COMCIDP    4876    A
COMCIDP  *         ENTRY  (B4) = 0TR, WHERE                                             COMCIDP    4877    A
COMCIDP  *                           T IS REGISTER TYPE (B=0,A=1)                       COMCIDP    4878    A
COMCIDP  *                           R IS REGISTER NR (0-7)                             COMCIDP    4879    A
COMCIDP  *                 (B6) = FWA TO STORE                                          COMCIDP    4880    A
COMCIDP  *                                                                              COMCIDP    4881    A
COMCIDP  *         EXIT   (B4) = UNCHANGED                                              COMCIDP    4882    A
COMCIDP  *                (B6) = UNCHANGED                                              COMCIDP    4883    A
COMCIDP  *                 (A6) = ADDR OF FULL ZERO WORD EOL MARKER                     COMCIDP    4884    A
COMCIDP  *                                                                              COMCIDP    4885    A
COMCIDP  *         USES   X - ALL     (INCLUDES ALL CALLS)                              COMCIDP    4886    A
COMCIDP  *                A - ALL BUT A0                                               COMCIDP    4887    A
COMCIDP  *                B - 2,3,4,5                                                  COMCIDP    4888    A
COMCIDP  *                                                                              COMCIDP    4889    A
COMCIDP  *         CALLS  CHK,COD,WOD                                                   COMCIDP    4890    A
COMCIDP                                                                                 COMCIDP    4891    A
COMCIDP                                                                                 COMCIDP    4892    A
COMCIDP   FAB      SUBR               ** ENTRY/EXIT **                                  COMCIDP    4893    A
COMCIDP            SX1    B4                                                            COMCIDP    4894    A
COMCIDP            MX0    -3                                                            COMCIDP    4895    A
COMCIDP            SA5    IDP=SVB+B4  (X5) = SAVED (TR)   (T=REG TYPE,R=REG NR)         COMCIDP    4896    A
COMCIDP            BX6    X0*X1       (X6) = REG TYPE (B=00B,A=10B)                     COMCIDP    4897    A
COMCIDP            SB5    B4          SAVE (B4) = 0TR                                   COMCIDP    4898    A
COMCIDP            BX7    -X0*X1      (X7) = REG NR (0-7)                               COMCIDP    4899    A
COMCIDP            LX6    -3                                                            COMCIDP    4900    A
COMCIDP            SA2    FABA+X6                                                       COMCIDP    4901    A
COMCIDP            SA3    FABB+X6                                                       COMCIDP    4902    A
COMCIDP            LX7    2*CHAR-1*CHAR                                                 COMCIDP    4903    A
COMCIDP            IX6    X2+X7       (X6) = .......TR.   (.=BLANK,T=RTYPE,R=RNR)       COMCIDP    4904    A
COMCIDP            LX7    5*CHAR-2*CHAR                                                 COMCIDP    4905    A
COMCIDP            SA6    B6                                                            COMCIDP    4906    A
COMCIDP            MX0    -18                                                           COMCIDP    4907    A
COMCIDP            IX7    X3+X7       (X7) = ..C(TR).=.                                 COMCIDP    4908    A
COMCIDP            BX1    -X0*X5                                                        COMCIDP    4909    A
COMCIDP            SA7    B6+2                                                          COMCIDP    4910    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP          RJ     =XCOD       CONVERT BINARY TO OCTAL DPC                COMCIDP    4911    A
COMCIDP          LX6    9*CHAR-6*CHAR                                          COMCIDP    4912    A
COMCIDP          SB4    B5          RESTORE (B4)                               COMCIDP    4913    A
COMCIDP          SA6    A6+B1                                                  COMCIDP    4914    A
COMCIDP          SB2    X5                                                     COMCIDP    4915    A
COMCIDP          RJ     CHK         CHECK CM ADDRESS                           COMCIDP    4916    A
COMCIDP          PL     B2,FAB2     IF ADDR OK                                 COMCIDP    4917    A
COMCIDP          SA1    =20H ** OUT OF RANGE **                                COMCIDP    4918    A
COMCIDP          SA2    A1+B1                                                  COMCIDP    4919    A
COMCIDP          BX6    X1                                                     COMCIDP    4920    A
COMCIDP          LX7    X2                                                     COMCIDP    4921    A
COMCIDP          EQ     FAB3                                                   COMCIDP    4922    A
COMCIDP                                                                        COMCIDP    4923    A
COMCIDP  *       HERE TO CONVERT C(TR) TO OCTAL DPC.                            COMCIDP    4924    A
COMCIDP                                                                        COMCIDP    4925    A
COMCIDP   FAB2   SA1    B2          (X1) = C(TR)                               COMCIDP    4926    A
COMCIDP          RJ     WOD         CONVERT A FULL BINARY WORD TO OCTAL DPC    COMCIDP    4927    A
COMCIDP                                                                        COMCIDP    4928    A
COMCIDP   FAB3   SA6    A7+B1                                                  COMCIDP    4929    A
COMCIDP          SA7    A6+B1                                                  COMCIDP    4930    A
COMCIDP          MX6    0                                                      COMCIDP    4931    A
COMCIDP          SA6    A7+B1       MARK EOL                                   COMCIDP    4932    A
COMCIDP          EQ     EXIT.                                                  COMCIDP    4933    A
COMCIDP                                                                        COMCIDP    4934    A
COMCIDP                                                                        COMCIDP    4935    A
COMCIDP   FABA   DATA   10H       B0  NNNNNN                                   COMCIDP    4936    A
COMCIDP          DATA   10H       A0  NNNNNN                                   COMCIDP    4937    A
COMCIDP   FABB   DATA   10H  C(B0) =  NNNNNNNNNNNNNNNNNNNN                      COMCIDP    4938    A
COMCIDP          DATA   10H  C(A0) =  NNNNNNNNNNNNNNNNNNNN                      COMCIDP    4939    A
COMCIDP   FLL    SPACE  4,10                                                    COMCIDP    4940    A
COMCIDP  **      FLL - CHECK FWA,LWA, AND LENGTH PARAMETERS.                    COMCIDP    4941    A
COMCIDP  *                                                                     COMCIDP    4942    A
COMCIDP  *                                                                     COMCIDP    4943    A
COMCIDP  *       ENTRY  (X2) = 18/0, 21/LEV OF IND ADDR, 21/FWA                COMCIDP    4944    A
COMCIDP  *              (X3) = 18/0, 21/LEV OF IND ADDR, 21/LWA                COMCIDP    4945    A
COMCIDP  *                     (X3) IS OPTIONAL                                COMCIDP    4946    A
COMCIDP  *              (X4) = 18/0, 21/LEV OF IND ADDR, 21/LEN                COMCIDP    4947    A
COMCIDP  *                     NOTE -- IF LEN = 0, DMP IS FROM FWA TO LWA,     COMCIDP    4948    A
COMCIDP  *                                 ELSE DMP IS FROM FWA TO FWA+LEN     COMCIDP    4949    A
COMCIDP  *                                                                     COMCIDP    4950    A
COMCIDP  *       EXIT   (B2) = .MI. IF FWA,LWA, OR LEN WAS BAD, ELSE .PL.      COMCIDP    4951    A
COMCIDP  *                                                                     COMCIDP    4952    A
COMCIDP  *              IF FWA,LWA,LEN IS OK, (B2) .PL. --                     COMCIDP    4953    A
COMCIDP  *                                                                     COMCIDP    4954    A
COMCIDP  *              (X2) = FWA.                                            COMCIDP    4955    A
COMCIDP  *              (X3) = LWA.                                            COMCIDP    4956    A
COMCIDP  *              (X4) = LEN.                                            COMCIDP    4957    A
COMCIDP  *              (AP=FWA) = FWA                                         COMCIDP    4958    A
COMCIDP  *              (AP=LWA) = LWA                                         COMCIDP    4959    A
COMCIDP  *              (AP=LEN) = LEN                                         COMCIDP    4960    A
COMCIDP  *                                                                     COMCIDP    4961    A
COMCIDP  *       USES   X - ALL BUT X5                                         COMCIDP    4962    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *                A - 1,2,6                                          COMCIDP    4963    A
COMCIDP  *                B - 2,3                                            COMCIDP    4964    A
COMCIDP  *                                                                   COMCIDP    4965    A
COMCIDP  *          CALLS  CHK,GIL                                           COMCIDP    4966    A
COMCIDP                                                                      COMCIDP    4967    A
COMCIDP                                                                      COMCIDP    4968    A
COMCIDP   FLL       SUBR                ** ENTRY/EXIT **                     COMCIDP    4969    A
COMCIDP                                                                      COMCIDP    4970    A
COMCIDP  *          CHECK FWA.                                               COMCIDP    4971    A
COMCIDP                                                                      COMCIDP    4972    A
COMCIDP             LX2    -21                                               COMCIDP    4973    A
COMCIDP             SB3    X2+          (B3) = LEVEL OF INDIRECT ADDRESSING  COMCIDP    4974    A
COMCIDP             LX2    21           RESTORE (X2)                         COMCIDP    4975    A
COMCIDP             RJ     GIL          GENERATE INDIRECT LOAD (IF NECESSARY) COMCIDP   4976    A
COMCIDP             MI     B2,EXIT.     IF FWA IS BAD                        COMCIDP    4977    A
COMCIDP             SX6    B2           SAVE (B2) = FWA                      COMCIDP    4978    A
COMCIDP             SA6    AP=FWA                                            COMCIDP    4979    A
COMCIDP                                                                      COMCIDP    4980    A
COMCIDP  *          CHECK LENGTH.                                            COMCIDP    4981    A
COMCIDP                                                                      COMCIDP    4982    A
COMCIDP   FLL2      MX0    -42                                               COMCIDP    4983    A
COMCIDP             BX2    -X0*X4                                            COMCIDP    4984    A
COMCIDP             ZR     X2,FLL3      IF NO LENGTH, USE LWA                COMCIDP    4985    A
COMCIDP             LX2    -21                                               COMCIDP    4986    A
COMCIDP             SB3    X2                                                COMCIDP    4987    A
COMCIDP             LX2    21                                                COMCIDP    4988    A
COMCIDP             RJ     GIL          GENERATE INDIRECT LOAD (IF NECESSARY) COMCIDP   4989    A
COMCIDP             MI     B2,EXIT.     IF LENGTH IS BAD                     COMCIDP    4990    A
COMCIDP             SX1    X6+B2        (X1) = FWA + LEN                     COMCIDP    4991    A
COMCIDP             SX6    B2           (X6) = LEN                           COMCIDP    4992    A
COMCIDP             SB2    X1-1         (B2) = LWA                           COMCIDP    4993    A
COMCIDP             SA6    AP=LEN                                            COMCIDP    4994    A
COMCIDP             RJ     CHK          CHECK CM ADDR                        COMCIDP    4995    A
COMCIDP             MI     B2,EXIT.     IF LWA IS BAD                        COMCIDP    4996    A
COMCIDP             SX6    B2+          SAVE (B2) = LWA                      COMCIDP    4997    A
COMCIDP             SA6    AP=LWA                                            COMCIDP    4998    A
COMCIDP             EQ     FLL4                                              COMCIDP    4999    A
COMCIDP                                                                      COMCIDP    5000    A
COMCIDP  *          CHECK LWA.                                               COMCIDP    5001    A
COMCIDP                                                                      COMCIDP    5002    A
COMCIDP   FLL3      BX2    X3                                                COMCIDP    5003    A
COMCIDP             LX2    -21                                               COMCIDP    5004    A
COMCIDP             BX7    -X0*X3                                            COMCIDP    5005    A
COMCIDP             SB3    X2                                                COMCIDP    5006    A
COMCIDP             SX4    B1           (X4) = DEFAULT LEN IS 1, IF LEN AND LWA =0 COMCIDP 5007  A
COMCIDP             LX2    21                                                COMCIDP    5008    A
COMCIDP             ZR     X7,FLL2      IF NO LWA, DEFAULT LEN=1             COMCIDP    5009    A
COMCIDP             RJ     GIL          GENERATE INDIRECT LOAD (IF NECESSARY) COMCIDP   5010    A
COMCIDP             MI     B2,EXIT.     IF LWA IS BAD                        COMCIDP    5011    A
COMCIDP             SA1    AP=FWA                                            COMCIDP    5012    A
COMCIDP             SX6    B2           SAVE (B2) = LWA                      COMCIDP    5013    A
COMCIDP             IX2    X6-X1                                             COMCIDP    5014    A

            0         1         2         3         4         5         6         7         8
            12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            SA6    A6+1                                                  COMCIDP     5015     A
COMCIDP            SX6    X2+1        (X6) = LEN                                COMCIDP     5016     A
COMCIDP            SA6    A6+1                                                  COMCIDP     5017     A
COMCIDP                                                                          COMCIDP     5018     A
COMCIDP  *         SET UP EXIT CONDITIONS.                                       COMCIDP     5019     A
COMCIDP                                                                          COMCIDP     5020     A
COMCIDP   FLL4     SA2    AP=FWA      (X2) = FWA                                COMCIDP     5021     A
COMCIDP            SA1    A2+B1                                                 COMCIDP     5022     A
COMCIDP            BX3    X1          (X3) = LWA                                COMCIDP     5023     A
COMCIDP            SA1    A1+B1                                                 COMCIDP     5024     A
COMCIDP            BX4    X1          (X4) = LEN                                COMCIDP     5025     A
COMCIDP            EQ     EXIT.                                                 COMCIDP     5026     A
COMCIDP   FOF      SPACE  4,10                                                  COMCIDP     5027     A
COMCIDP   .IDPOS   IFNE   .OS,2                                                 CCGA018       24     A
COMCIDP  **        FOF - FLUSH OUTPUT FILE.                                     COMCIDP     5028     A
COMCIDP  *                                                                      COMCIDP     5029     A
COMCIDP  *                                                                      COMCIDP     5030     A
COMCIDP  *         *FOF* CONDITIONALLY FLUSHES AN OUTPUT FILE, BASED ON         COMCIDP     5031     A
COMCIDP  *         WHETHER THERE IS ANYTHING IN THE BUFFER FOR THAT FILE,       COMCIDP     5032     A
COMCIDP  *         I.E. IF *IN .NE. OUT*.                                       COMCIDP     5033     A
COMCIDP  *                                                                      COMCIDP     5034     A
COMCIDP  *         ENTRY  (B2) = FWA OF FET.                                    COMCIDP     5035     A
COMCIDP  *                                                                      COMCIDP     5036     A
COMCIDP  *         EXIT   NONE                                                  COMCIDP     5037     A
COMCIDP  *                                                                      COMCIDP     5038     A
COMCIDP  *         USES   ALL BUT A0,X0,A5,X5     (INCLUDES ALL CALLS)          COMCIDP     5039     A
COMCIDP  *                                                                      COMCIDP     5040     A
COMCIDP  *         CALLS  WRITER                                                COMCIDP     5041     A
COMCIDP                                                                          COMCIDP     5042     A
COMCIDP                                                                          COMCIDP     5043     A
COMCIDP   FOF      SUBR               ** ENTRY/EXIT **                          COMCIDP     5044     A
COMCIDP            MI     B2,EXIT.    IF FET NOT DEFINED                        COMCIDP     5045     A
COMCIDP            SA1    B2+2        (X1) = IN                                 COMCIDP     5046     A
COMCIDP            SA2    A1+B1       (X2) = OUT                                COMCIDP     5047     A
COMCIDP            IX7    X1-X2                                                 COMCIDP     5048     A
COMCIDP            ZR     X7,EXIT.    IF IN = OUT, NOTHING TO FLUSH...          COMCIDP     5049     A
COMCIDP            WRITER B2,,RCL                                               COMCIDP     5050     A
COMCIDP            EQ     EXIT.                                                 COMCIDP     5051     A
COMCIDP   .IDPOS   ENDIF                                                        CCGA018       25     A
COMCIDP   FRA      SPACE  4,10                                                  COMCIDP     5052     A
COMCIDP  **        FRA - FIND RELATIVE ADDRESS.                                 COMCIDP     5053     A
COMCIDP  *                                                                      COMCIDP     5054     A
COMCIDP  *                                                                      COMCIDP     5055     A
COMCIDP  *         *FRA* IS USED THROUGHOUT IDP WHENEVER HE WOULD LIKE TO       COMCIDP     5056     A
COMCIDP  *         OUTPUT AN ABSOLUTE ADDRESS IN A DECK-RELATIVE FORM (FOR      COMCIDP     5057     A
COMCIDP  *         READABILITY/UNDERSTANDABILITY BY A HUMAN).                   COMCIDP     5058     A
COMCIDP  *                                                                      COMCIDP     5059     A
COMCIDP  *         *FRA* TRIES TO PRODUCE A STRING OF THE FORM--                COMCIDP     5060     A
COMCIDP  *                                                                      COMCIDP     5061     A
COMCIDP  *                NNNNNN.IN.XXXXXXXX00                                  COMCIDP     5062     A
COMCIDP  *                                                                      COMCIDP     5063     A
COMCIDP  *         WHERE  N = CONVERTED DPC RELATIVE ADDR DIGITS.               COMCIDP     5064     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

COMCIDP  *                X = DECK NAME.                                    COMCIDP    5065    A
COMCIDP  *                0 = 00B, I.E. EOL BITS.                           COMCIDP    5066    A
COMCIDP  *                . = 55B, I.E. BLANKS.                             COMCIDP    5067    A
COMCIDP  *                                                                  COMCIDP    5068    A
COMCIDP  *         FOR EXAMPLE, GIVEN A DECK CALLED *IDP* THAT HAS ITS FWA AT  COMCIDP    5069    A
COMCIDP  *         ABS ADDR 4000, THEN A CALL TO *FRA* WITH ABS ADDR 4222   COMCIDP    5070    A
COMCIDP  *         MIGHT PRODUCE --                                         COMCIDP    5071    A
COMCIDP  *                                                                  COMCIDP    5072    A
COMCIDP  *                222    IN IDP                                     COMCIDP    5073    A
COMCIDP  *                                                                  COMCIDP    5074    A
COMCIDP  *         *FRA* TRIES TO DO HIS JOB IN THE FOLLOWING WAYS--        COMCIDP    5075    A
COMCIDP  *                                                                  COMCIDP    5076    A
COMCIDP  *           1. IF THE USER PROVIDED AN *FRA=* (VIA =Y), THEN THIS *FRA*  COMCIDP    5077    A
COMCIDP  *              WILL CALL IT.  THAT IS, THE USER'S *FRA=* TAKES     COMCIDP    5078    A
COMCIDP  *              PRECEDENCE.                                         COMCIDP    5079    A
COMCIDP  *                                                                  COMCIDP    5080    A
COMCIDP  *           2. IF THE USER PROVIDED AN *RNA* (ROUTINE NAME/ADDRESS)  COMCIDP    5081    A
COMCIDP  *              TABLE, THEN THIS *FRA* WILL USE/SEARCH IT.          COMCIDP    5082    A
COMCIDP  *                                                                  COMCIDP    5083    A
COMCIDP  *              *FRA* DETECTS THE PRESENCE/ABSENCE OF AN *RNA* TABLE  COMCIDP    5084    A
COMCIDP  *              VIA THE PRESENCE/ABSENCE OF THE CELL *RPV=RNA* (VIA =Y).  COMCIDP    5085    A
COMCIDP  *              IF *RPV=RNA* IS PROVIDED, THEN IT IS A CELL THAT    COMCIDP    5086    A
COMCIDP  *              **CONTAINS** THE FWA OF THE *RNA* TABLE.            COMCIDP    5087    A
COMCIDP  *                                                                  COMCIDP    5088    A
COMCIDP  *              NOTE THAT THE SYMBOL NAME *RPV=RNA* IS FOR COMPATABILITY  COMCIDP    5089    A
COMCIDP  *              WITH *CCOMRPV* (COMPILER COMMON REPRIEVE PROCESSOR).  COMCIDP    5090    A
COMCIDP  *                                                                  COMCIDP    5091    A
COMCIDP  *              AN *RNA* TABLE HAS THE FOLLOWING FORM--             COMCIDP    5092    A
COMCIDP  *                                                                  COMCIDP    5093    A
COMCIDP  *                  42/0LDECK1,18/FWA OF DECK1                      COMCIDP    5094    A
COMCIDP  *                  42/0LDECK2,18/FWA OF DECK2                      COMCIDP    5095    A
COMCIDP  *                          .                                       COMCIDP    5096    A
COMCIDP  *                          .                                       COMCIDP    5097    A
COMCIDP  *                  42/0LDECKN,18/FWA OF DECKN                      COMCIDP    5098    A
COMCIDP  *                  60/0    END OF TABLE MARK                       COMCIDP    5099    A
COMCIDP  *                  60/LINK TO NEXT *RNA* TABLE, ELSE               COMCIDP    5100    A
COMCIDP  *                     .ZR. IF END OF ALL TABLES.                   COMCIDP    5101    A
COMCIDP  *                                                                  COMCIDP    5102    A
COMCIDP  *              *RNA* TABLES MAY BE LINKED TO ACCOMODATE OVERLAYED HOST  COMCIDP    5103    A
COMCIDP  *              PROGRAMS.                                           COMCIDP    5104    A
COMCIDP  *                                                                  COMCIDP    5105    A
COMCIDP  *              NOTE THAT *RNA* TABLE(S) DO **NOT** HAVE TO BE ORDERED  COMCIDP    5106    A
COMCIDP  *              IN DECK-ASCENDING ORDER.  *FRA* SEARCHS FOR THE     COMCIDP    5107    A
COMCIDP  *              **CLOSEST** OCCURENCE.                              COMCIDP    5108    A
COMCIDP  *                                                                  COMCIDP    5109    A
COMCIDP  *              NOTE ALSO THAT *FRA* CHECKS TO SEE IF (RPV=RNA) IS  COMCIDP    5110    A
COMCIDP  *              IN FL, VIA *CHK* (CHECK CM ADDRESS), BEFORE USING   COMCIDP    5111    A
COMCIDP  *              IT.  IN THIS WAY, THE HOST PROGRAM CAN IF IT WISHS  COMCIDP    5112    A
COMCIDP  *              SELECT/DESELECT *RNA* TABLES BY MERELY SETTING      COMCIDP    5113    A
COMCIDP  *              (RPV=RNA) OUT-OF-RANGE (E.G. .MI.).                 COMCIDP    5114    A
COMCIDP  *                                                                  COMCIDP    5115    A
COMCIDP  *           3. IF NEITHER OF THE ABOVE CONDITIONS ARE SATISFIED, THEN  COMCIDP    5116    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *              *FRA* DOESNT/CANT DO ANYTHING...                    COMCIDP    5117    A
COMCIDP  *                                                                  COMCIDP    5118    A
COMCIDP  *         ENTRY  (X1) = BINARY ABS ADDR TO FIND.                   COMCIDP    5119    A
COMCIDP  *                                                                  COMCIDP    5120    A
COMCIDP  *         ENTRY  IF *FRA* COULD FIND A DECK-RELATIVE ADDR--        COMCIDP    5121    A
COMCIDP  *                                                                  COMCIDP    5122    A
COMCIDP  *                (B7) = .PL. TO INDICATE THAT *FRA* FOUND A DECK   COMCIDP    5123    A
COMCIDP  *                           RELATIVE ADDRESS.                      COMCIDP    5124    A
COMCIDP  *                (X6) = NNNNNN.IN.                                 COMCIDP    5125    A
COMCIDP  *                (X7) = XXXXXXXX00     (SEE PREAMBLE ABOVE FOR DETAILS)  COMCIDP  5126  A
COMCIDP  *                                                                  COMCIDP    5127    A
COMCIDP  *                IF *FRA* COULD NOT FIND A DECK RELATIVE ADDR--    COMCIDP    5128    A
COMCIDP  *                                                                  COMCIDP    5129    A
COMCIDP  *                (B7) = .MI. TO INDICATE THAT *FRA* COULD NOT      COMCIDP    5130    A
COMCIDP  *                           FIND A DECK RELATIVE ADDRESS.          COMCIDP    5131    A
COMCIDP  *                (X1) = AS ON ENTRY.                               COMCIDP    5132    A
COMCIDP  *                (X6) = +0.                                        COMCIDP    5133    A
COMCIDP  *                (X7) = +0.                                        COMCIDP    5134    A
COMCIDP  *                                                                  COMCIDP    5135    A
COMCIDP  *         USES   ALL BUT A0,X0,A5,X5,A6                            COMCIDP    5136    A
COMCIDP  *                                                                  COMCIDP    5137    A
COMCIDP  *         CALLS  CHK,COD,FRA=(IF AVAIL)                            COMCIDP    5138    A
COMCIDP                                                                     COMCIDP    5139    A
COMCIDP                                                                     COMCIDP    5140    A
COMCIDP   FRA      SUBR              ** ENTRY/EXIT **                       COMCIDP    5141    A
COMCIDP            SB6    X1+        SAVE (X1)                              COMCIDP    5142    A
COMCIDP            SB7    =YFRA=                                            COMCIDP    5143    A
COMCIDP            BX6    X6-X6                                             COMCIDP    5144    A
COMCIDP            MX7    0                                                 COMCIDP    5145    A
COMCIDP            MI     B7,FRA1    IF USER DID NOT PROVIDE *FRA=*         COMCIDP    5146    A
COMCIDP            SB7    EXIT.      (B7) = RETURN ADDR FOR *FRA=*          COMCIDP    5147    A
COMCIDP            EQ     =YFRA=     FIND RELATIVE ADDRESS                  COMCIDP    5148    A
COMCIDP                                                                     COMCIDP    5149    A
COMCIDP  *         HERE TO TRY *RNA* TABLE(S).                              COMCIDP    5150    A
COMCIDP                                                                     COMCIDP    5151    A
COMCIDP   FRA1     SB2    =YRPV=RNA                                         COMCIDP    5152    A
COMCIDP            RJ     CHK        CHECK CM ADDRESS                       COMCIDP    5153    A
COMCIDP            SB7    B2+                                               COMCIDP    5154    A
COMCIDP            MI     B7,EXIT.   IF USER DID NOT PROVIDE *RNA* TABLE    COMCIDP    5155    A
COMCIDP                                                                     COMCIDP    5156    A
COMCIDP            SA2    =YRPV=RNA  (X2) = FWA OF 1ST RNA TABLE            COMCIDP    5157    A
COMCIDP            SB2    B6         (B2) = ADDRESS RELATIVE TO RA+0        COMCIDP    5158    A
COMCIDP            SA2    X2         (A2,X2) = A+C OF 1ST ENTRY IN RNA TABLE COMCIDP   5159    A
COMCIDP            SB3    X2                                                COMCIDP    5160    A
COMCIDP            SB4    B0                                                COMCIDP    5161    A
COMCIDP            SB5    A2+                                               COMCIDP    5162    A
COMCIDP                                                                     COMCIDP    5163    A
COMCIDP  *         SEARCH ROUTINE NAME ADDRESS TABLE.                       COMCIDP    5164    A
COMCIDP                                                                     COMCIDP    5165    A
COMCIDP   FRA2     BSS                                                      COMCIDP    5166    A
COMCIDP            ZR     X2,FRA3    IF END OF BLOCK                        COMCIDP    5167    A
COMCIDP            SB3    X2                                                COMCIDP    5168    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            SA2    A2+B1       (X2) = NEXT TABLE ENTRY                          COMCIDP    5169    A
COMCIDP            GT     B3,B2,FRA2  IF CURRENT ENTRY BEYOND ABS ADDR                 COMCIDP    5170    A
COMCIDP            GE     B4,B3,FRA2  IF CURRENT NOT CLOSER                            COMCIDP    5171    A
COMCIDP            SB5    A2-B1       CLOSEST SO FAR                                   COMCIDP    5172    A
COMCIDP            SB4    B3                                                           COMCIDP    5173    A
COMCIDP            EQ     FRA2        KEEP LOOKING                                     COMCIDP    5174    A
COMCIDP                                                                                COMCIDP    5175    A
COMCIDP  *         CHECK FOR TABLE CONTINUATION LINK.                                  COMCIDP    5176    A
COMCIDP                                                                                COMCIDP    5177    A
COMCIDP   FRA3     BSS                                                                 COMCIDP    5178    A
COMCIDP            SA2    A2+B1       END OF TABLE/TABLE LINK WORD                     COMCIDP    5179    A
COMCIDP            ZR     X2,FRA4     IF END OF TABLE                                  COMCIDP    5180    A
COMCIDP            SA2    X2          (A2,X2) = A+C OF 1ST WORD IN NEXT BLOCK          COMCIDP    5181    A
COMCIDP            EQ     FRA2        CONTINUE TABLE SEARCH...                         COMCIDP    5182    A
COMCIDP                                                                                COMCIDP    5183    A
COMCIDP  *         EXTRACT ROUTINE NAME AND FORMAT ABSOLUTE ADDRESS.                   COMCIDP    5184    A
COMCIDP                                                                                COMCIDP    5185    A
COMCIDP   FRA4     BSS                                                                 COMCIDP    5186    A
COMCIDP            SX1    B2-B4       (X1) = BIN ADDR, RELATIVE TO CLOSEST FIND        COMCIDP    5187    A
COMCIDP            RJ     =XCOD       CONVERT TO OCTAL DISPLAY CODE                    COMCIDP    5188    A
COMCIDP            SA1    B5          (X1) = 42/0LNAME,18/ADDRESS                      COMCIDP    5189    A
COMCIDP            MX7    42                                                           COMCIDP    5190    A
COMCIDP            SX3    2R  &2RIN                                                    COMCIDP    5191    A
COMCIDP            BX7    X7*X1       (X7) = ROUTINE NAME, 0L FORMAT                   COMCIDP    5192    A
COMCIDP            LX3    6                                                            COMCIDP    5193    A
COMCIDP            BX6    X4-X3       (X6) = RELATIVE ADDRESS, *NNNNNN IN *            COMCIDP    5194    A
COMCIDP            EQ     EXIT.                                                        COMCIDP    5195    A
COMCIDP   FRK      SPACE  4,10                                                         COMCIDP    5196    A
COMCIDP  **        FRK - CHECK FREQUENCY PARAMETERS.                                   COMCIDP    5197    A
COMCIDP  *                                                                             COMCIDP    5198    A
COMCIDP  *                                                                             COMCIDP    5199    A
COMCIDP  *         ENTRY  (X5)   =  FWA OF SNAP PARAMETER LIST                         COMCIDP    5200    A
COMCIDP  *                                                                             COMCIDP    5201    A
COMCIDP  *         FREQUENCY PARAMETER LIST AT (X5)+SN=FRK --                          COMCIDP    5202    A
COMCIDP  *                                                                             COMCIDP    5203    A
COMCIDP  *         VFD    60/LOWER LIMIT (LL)                                          COMCIDP    5204    A
COMCIDP  *         VFD    60/UPPER LIMIT (UL)                                          COMCIDP    5205    A
COMCIDP  *         VFD    60/INCREMENT  (INC)                                          COMCIDP    5206    A
COMCIDP  *         VFD    60/10HNAME                                                   COMCIDP    5207    A
COMCIDP  *         VFD    60/SNAP COUNT                                                COMCIDP    5208    A
COMCIDP  *                                                                             COMCIDP    5209    A
COMCIDP  *         EXIT   (X5)   =  .ZR. IF NO SNAP THIS TIME, ELSE UNCHANGED          COMCIDP    5210    A
COMCIDP  *                                                                             COMCIDP    5211    A
COMCIDP  *         USES   X - ALL     (BUT X5 IF SNAP THIS TIME)                       COMCIDP    5212    A
COMCIDP  *                A - 1,2,3,4                                                 COMCIDP    5213    A
COMCIDP  *                B - 7                                                       COMCIDP    5214    A
COMCIDP  *                                                                             COMCIDP    5215    A
COMCIDP  *         CALLS  NONE                                                         COMCIDP    5216    A
COMCIDP                                                                                COMCIDP    5217    A
COMCIDP                                                                                COMCIDP    5218    A
COMCIDP   FRK.NO   SX5    0                                                            COMCIDP    5219    A
COMCIDP                                                                                COMCIDP    5220    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP   FRK      SUBR                ** ENTRY/EXIT **                          COMCIDP     5221     A
COMCIDP            SA1    X5+SN=LL     (X1) = LL                                 COMCIDP     5222     A
COMCIDP            SA2    A1+B1        (X2) = UL                                 COMCIDP     5223     A
COMCIDP            SA3    A2+B1        (X3) = INC                                COMCIDP     5224     A
COMCIDP            SA4    X5+SN=CNT    (X4) = SNAP COUNT                         COMCIDP     5225     A
COMCIDP            MI     X1,FRK.NO    IF LL BAD                                 COMCIDP     5226     A
COMCIDP            MI     X2,FRK.NO    IF UL BAD                                 COMCIDP     5227     A
COMCIDP            MI     X3,FRK.NO     IF INC BAD                               COMCIDP     5228     A
COMCIDP            SX6    X4+B1        (X6) = SNAP COUNT + 1                     COMCIDP     5229     A
COMCIDP            SA6    A4+                                                    COMCIDP     5230     A
COMCIDP            IX7    X6-X1        (X7) = SNAP CNT - LOWER LIMIT             COMCIDP     5231     A
COMCIDP            MI     X7,FRK.NO    IF NOT TIME TO SNAP YET                   COMCIDP     5232     A
COMCIDP            IX7    X2-X6        (X7) = UPPER LIMIT - SNAP COUNT           COMCIDP     5233     A
COMCIDP            PX3    X3                                                     COMCIDP     5234     A
COMCIDP            MI     X7,FRK.NO    IF PAST UPPER LIMIT                       COMCIDP     5235     A
COMCIDP                                                                          COMCIDP     5236     A
COMCIDP   *        COMPUTE  (COUNT/INC)*INC-COUNT.                               COMCIDP     5237     A
COMCIDP                                                                          COMCIDP     5238     A
COMCIDP            PX6    X6                                                     COMCIDP     5239     A
COMCIDP            NX3    X3                                                     COMCIDP     5240     A
COMCIDP            NX6    X6                                                     COMCIDP     5241     A
COMCIDP            FX7    X6/X3                                                  COMCIDP     5242     A
COMCIDP            UX7    X7,B7                                                  COMCIDP     5243     A
COMCIDP            LX7    X7,B7                                                  COMCIDP     5244     A
COMCIDP            PX7    X7                                                     COMCIDP     5245     A
COMCIDP            NX7    X7                                                     COMCIDP     5246     A
COMCIDP            FX7    X7*X3                                                  COMCIDP     5247     A
COMCIDP            IX7    X7-X6                                                  COMCIDP     5248     A
COMCIDP            NZ     X7,FRK.NO    IF NOT AT A SNAP INCREMENT                COMCIDP     5249     A
COMCIDP            EQ     EXIT.        SNAP SHOULD BE HONORED...                 COMCIDP     5250     A
COMCIDP   FRZ      SPACE  4,10                                                   COMCIDP     5251     A
COMCIDP   .IDPOS   IFNE   .OS,2                                                  CCGA018       26     A
COMCIDP   **       FRZ - FREEZE INTERACTIVE SESSION.                             COMCIDP     5252     A
COMCIDP   *                                                                      COMCIDP     5253     A
COMCIDP   *                                                                      COMCIDP     5254     A
COMCIDP   *        PERFORMS A CHECKPOINT OF AN INTERACTIVE SESSION BY            COMCIDP     5255     A
COMCIDP   *        WRITING THE CURRENT CORE IMAGE TO LFN *F.FRZ* IN A            COMCIDP     5256     A
COMCIDP   *        SPECIAL FORMAT.                                               COMCIDP     5257     A
COMCIDP   *                                                                      COMCIDP     5258     A
COMCIDP   *        *F.FRZ* FORMAT --                                             COMCIDP     5259     A
COMCIDP   *                                                                      COMCIDP     5260     A
COMCIDP   *          RECORD 1 CONSISTS OF A LOADER ABSOLUTE BINARY, CALLED *IFR* COMCIDP     5261     A
COMCIDP   *          (IDP FREEZE RESTART), THAT WHEN EXECUTED WILL READ THE      COMCIDP     5262     A
COMCIDP   *          FROZEN HOST BACK INTO CORE (OVERLAYING ITSELF), AND         COMCIDP     5263     A
COMCIDP   *          THEN REINVOKE *IDP*.                                        COMCIDP     5264     A
COMCIDP   *                                                                      COMCIDP     5265     A
COMCIDP   *          RECORD 2 CONTAINS THE CORE IMAGE OF THE FROZEN HOST.        COMCIDP     5266     A
COMCIDP   *                                                                      COMCIDP     5267     A
COMCIDP   *        SEE ROUTINE *IFR* IN DATA SECTION BEGINNING AT ADDR *FW.IFR*. COMCIDP     5268     A
COMCIDP   *                                                                      COMCIDP     5269     A
COMCIDP   *        ENTRY  NONE                                                   COMCIDP     5270     A
COMCIDP   *                                                                      COMCIDP     5271     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *         EXIT   *F.FRZ* WRITTEN                                    COMCIDP    5272    A
COMCIDP  *                                                                   COMCIDP    5273    A
COMCIDP  *         USES   ALL BUT A0,X0,A5,X5   (INCLUDES ALL CALLS)         COMCIDP    5274    A
COMCIDP  *                                                                   COMCIDP    5275    A
COMCIDP  *         CALLS  MEMORY,REWIND,WRITER,WRITEW                        COMCIDP    5276    A
COMCIDP                                                                      COMCIDP    5277    A
COMCIDP                                                                      COMCIDP    5278    A
COMCIDP   FRZ      SUBR               ** ENTRY/EXIT **                       COMCIDP    5279    A
COMCIDP                                                                      COMCIDP    5280    A
COMCIDP  *         FIND OUT ACTUAL FIELD LENGTH SCM (AFLS) SO THAT *IFR*     COMCIDP    5281    A
COMCIDP  *         CAN KNOW HOW MUCH IS NEEDED TO RESTART.  IF *CMM* IS ACTIVE,  COMCIDP  5282  A
COMCIDP  *         SAVE THIS KNOWLEDGE SO THAT *IFR* CAN INDICATE SO         COMCIDP    5283    A
COMCIDP  *         TO OPERATING SYSTEM.                                      COMCIDP    5284    A
COMCIDP                                                                      COMCIDP    5285    A
COMCIDP            SA1    RA.LWP                                             COMCIDP    5286    A
COMCIDP            BX6    X6-X6                                              COMCIDP    5287    A
COMCIDP            SX7    X1          (X7) = .MI. IF *CMM* ACTIVE, ELSE .PL. COMCIDP    5288    A
COMCIDP            SA6    MEM#HOST                                           COMCIDP    5289    A
COMCIDP            MEMORY SCM,A6,RCL                                         COMCIDP    5290    A
COMCIDP            SX2    B1                                                 COMCIDP    5291    A
COMCIDP            AX7    59          (X7) = -0 IF *CMM* ACTIVE, ELSE +0     COMCIDP    5292    A
COMCIDP            SA1    MEM#HOST                                           COMCIDP    5293    A
COMCIDP            BX7    X2*X7       (X7) = 1 IF *CMM* ACTIVE, ELSE +0      COMCIDP    5294    A
COMCIDP            LX7    2-0                                                COMCIDP    5295    A
COMCIDP            SX7    X7+B1       SET BIT 1S0                            COMCIDP    5296    A
COMCIDP            BX6    X1-X7       SET *CMM* IF APPROPRIATE/CLEAR COMPLETE BIT  COMCIDP  5297  A
COMCIDP            SA6    A1                                                 COMCIDP    5298    A
COMCIDP                                                                      COMCIDP    5299    A
COMCIDP  *         NEED TO SAVE 3 SPECIAL WORDS BECAUSE WE ARE ONLY WRITING  COMCIDP    5300    A
COMCIDP  *         FROM *RA.MTR+1* TO FL-2 TO *F.FRZ*.                       COMCIDP    5301    A
COMCIDP  *           1. SAVE (RA.SSW) BECAUSE APPARENTLY *CIO* WILL SOMETIMES COMCIDP   5302    A
COMCIDP  *              NOT ALLOW ONE TO HAVE *RA.MTR* IN CIO BUFFER AREA.     COMCIDP  5303    A
COMCIDP  *           2. SAVE WORDS AT FL-1 AND FL-2 BECAUSE LWA AND LWA+1 OF CIO COMCIDP  5304  A
COMCIDP  *              BUFFER ARE NOT WRITTEN.                                COMCIDP  5305    A
COMCIDP                                                                      COMCIDP    5306    A
COMCIDP            AX1    30          (X1) = AFLS OF FROZEN HOST             COMCIDP    5307    A
COMCIDP            SA2    RA.SSW                                             COMCIDP    5308    A
COMCIDP            SA3    X1-2                                               COMCIDP    5309    A
COMCIDP            SA4    A3+B1                                              COMCIDP    5310    A
COMCIDP            BX6    X2                                                 COMCIDP    5311    A
COMCIDP            LX7    X3                                                 COMCIDP    5312    A
COMCIDP            SA6    FRZ#SAV                                            COMCIDP    5313    A
COMCIDP            SA7    A6+B1                                              COMCIDP    5314    A
COMCIDP            BX6    X4                                                 COMCIDP    5315    A
COMCIDP            SA6    A7+B1                                              COMCIDP    5316    A
COMCIDP                                                                      COMCIDP    5317    A
COMCIDP  *         SET UP AND *OPEN* *F.FRZ* FET FOR RECORD 1 WRITE.         COMCIDP    5318    A
COMCIDP  *         FIRST, THOUGH, WE NEED TO SET THE *COMPLETE* BIT          COMCIDP    5319    A
COMCIDP  *         IN *F.FRZ* FET BECAUSE IF WE ARE NOW FREEZING AN          COMCIDP    5320    A
COMCIDP  *         INTERACTIVE SESSION THAT WAS PREVIOUSLY FROZEN AND UNFROZEN, COMCIDP  5321  A
COMCIDP  *         THEN *F.FRZ* IS MARKED AS STILL BEING ACTIVE.  THIS HAPPENS COMCIDP  5322  A
COMCIDP  *         BECAUSE WHEN THE PREVIOUS SESSION WAS FROZEN, *F.FRZ* WAS COMCIDP  5323  A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *        USED TO WRITE ITSELF OUT, AND WAS THEREFORE STILL ACTIVE WHEN      COMCIDP    5324    A
COMCIDP  *        CORE IMAGE WAS WRITTEN TO DISK.                                    COMCIDP    5325    A
COMCIDP                                                                              COMCIDP    5326    A
COMCIDP           SA1    F.FRZ                                                       COMCIDP    5327    A
COMCIDP           MX2    -1                                                          COMCIDP    5328    A
COMCIDP           BX6    -X2+X1      SET COMPLETE BIT                                COMCIDP    5329    A
COMCIDP           SA6    A1                                                          COMCIDP    5330    A
COMCIDP                                                                              COMCIDP    5331    A
COMCIDP           SX6    FW.IFR                                                      COMCIDP    5332    A
COMCIDP           SX7    FW.IFR+L.IFR-1                                              COMCIDP    5333    A
COMCIDP           SA6    A6+B1       FIRST  = FW.IFR                                 COMCIDP    5334    A
COMCIDP           SA7    A6+B1       IN     = FW.IFR+L.IFR-1                         COMCIDP    5335    A
COMCIDP           SA6    A7+B1       OUT    = FW.IFR                                 COMCIDP    5336    A
COMCIDP           SX7    X7+B1                                                       COMCIDP    5337    A
COMCIDP           SA7    A6+B1       LIMIT  = FW.IFR+L.IFR                           COMCIDP    5338    A
COMCIDP                                                                              COMCIDP    5339    A
COMCIDP           OPEN   A1,,RCL                                                     COMCIDP    5340    A
COMCIDP           SA1    F.FRZ+4     (X1) = LIMIT                                    COMCIDP    5341    A
COMCIDP           SX6    X1-1                                                        COMCIDP    5342    A
COMCIDP           SA6    F.FRZ+2     IN = LIMIT-1                                    COMCIDP    5343    A
COMCIDP                                                                              COMCIDP    5344    A
COMCIDP  *        WRITE INTERACTIVE FREEZE RESTART PROGRAM, *IFR*,                   COMCIDP    5345    A
COMCIDP  *        TO RECORD 1 OF *F.FRZ*.                                            COMCIDP    5346    A
COMCIDP                                                                              COMCIDP    5347    A
COMCIDP           WRITER F.FRZ,,RCL                                                  COMCIDP    5348    A
COMCIDP                                                                              COMCIDP    5349    A
COMCIDP  *        SET UP *F.FRZ* FET SO THAT BUFFER IS RA.MTR+1 THRU FL-1.           COMCIDP    5350    A
COMCIDP                                                                              COMCIDP    5351    A
COMCIDP           SA1    MEM#HOST                                                    COMCIDP    5352    A
COMCIDP           SX6    RA.MTR+1    (X6) = FIRST = OUT                              COMCIDP    5353    A
COMCIDP           LX1    30                                                          COMCIDP    5354    A
COMCIDP           SX7    X1-2        (X7) = IN = AFLS-2 (FOR SAFETY)                 COMCIDP    5355    A
COMCIDP           SA6    F.FRZ+1     FIRST = RA.MTR+1                                COMCIDP    5356    A
COMCIDP           SA7    A6+B1       IN =AFLS-2                                      COMCIDP    5357    A
COMCIDP           SA6    A7+B1       OUT = RA.MTR+1                                  COMCIDP    5358    A
COMCIDP           SX7    X7+B1       (X7) = LIMIT = AFLS-1 (FOR SAFETY)              COMCIDP    5359    A
COMCIDP           SA7    A6+B1       LIMIT = AFLS-1                                  COMCIDP    5360    A
COMCIDP                                                                              COMCIDP    5361    A
COMCIDP  *        WRITE ALL OF SCM TO *F.FRZ/RECORD 2*.                              COMCIDP    5362    A
COMCIDP                                                                              COMCIDP    5363    A
COMCIDP           WRITER F.FRZ,,RCL                                                  COMCIDP    5364    A
COMCIDP           EQ     EXIT.                                                       COMCIDP    5365    A
COMCIDP                                                                              COMCIDP    5366    A
COMCIDP                                                                              COMCIDP    5367    A
COMCIDP   FRZ#SAV  =     APL         FWA OF SAVE AREA FOR (RA.SSW), (FL-2),         COMCIDP    5368    A
COMCIDP  *                           AND (FL-1)                                     COMCIDP    5369    A
COMCIDP   .IDPOS   ENDIF                                                             CCGA018      27    A
COMCIDP   GIL      SPACE  4,10                                                       COMCIDP    5370    A
COMCIDP  **        GIL - GENERATE INDIRECT LOAD.                                     COMCIDP    5371    A
COMCIDP  *                                                                           COMCIDP    5372    A
COMCIDP  *                                                                           COMCIDP    5373    A
COMCIDP  *                GENERATES 0 THRU N INDIRECT LOADS, CHECKING THE ADDRESS    COMCIDP    5374    A
```

```
          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *         TO BE LOADED BEFORE EACH LOAD.                                COMCIDP    5375    A
COMCIDP  *                                                                       COMCIDP    5376    A
COMCIDP  *         ENTRY  (X2) = ADDRESS (BITS 59-18) IGNORED)                   COMCIDP    5377    A
COMCIDP  *                (B3) = LEVEL OF INDIRECT ADDRESSING,                   COMCIDP    5378    A
COMCIDP  *                        (B3) = 0 MEANS DIRECT ADDRESSING               COMCIDP    5379    A
COMCIDP  *                                                                       COMCIDP    5380    A
COMCIDP  *         EXIT   (B2) = ADDRESS AT END OF INDIRECT CHAIN, IF (B2) .PL.  COMCIDP    5381    A
COMCIDP  *                      IF (B2) .MI., ONE OF THE ADDRESSES WAS BAD       COMCIDP    5382    A
COMCIDP  *                                                                       COMCIDP    5383    A
COMCIDP  *         USES   X - 1,2                                                COMCIDP    5384    A
COMCIDP  *                A - 1,2                                                COMCIDP    5385    A
COMCIDP  *                B - 2,3                                                COMCIDP    5386    A
COMCIDP  *                                                                       COMCIDP    5387    A
COMCIDP  *         CALLS  CHK                                                    COMCIDP    5388    A
COMCIDP                                                                          COMCIDP    5389    A
COMCIDP                                                                          COMCIDP    5390    A
COMCIDP   GIL      SUBR              ** ENTRY/EXIT **                            COMCIDP    5391    A
COMCIDP                                                                          COMCIDP    5392    A
COMCIDP   GIL2     SB2    X2                                                     COMCIDP    5393    A
COMCIDP            SB3    B3-B1                                                  COMCIDP    5394    A
COMCIDP            RJ     CHK         CHECK CM ADDRESS                           COMCIDP    5395    A
COMCIDP            MI     B2,EXIT.    IF ADDR IS BAD                             COMCIDP    5396    A
COMCIDP            SA2    B2+                                                    COMCIDP    5397    A
COMCIDP            GE     B3,B0,GIL2  IF MORE LEVELS OF INDIRECT ADDRESSING TO GO COMCIDP    5398    A
COMCIDP            EQ     EXIT.                                                  COMCIDP    5399    A
COMCIDP   HDR      SPACE  4,10                                                   COMCIDP    5400    A
COMCIDP  **        HDR - PRINT SNAP HEADER.                                      COMCIDP    5401    A
COMCIDP  *                                                                       COMCIDP    5402    A
COMCIDP  *                                                                       COMCIDP    5403    A
COMCIDP  *                PRINTS A SNAP HEADER OF THE FORM--                     COMCIDP    5404    A
COMCIDP  *                                                                       COMCIDP    5405    A
COMCIDP  * NAME      SNAP NR  NNNNNN    CALLED BY NNNNNN IN XXXXXXX              COMCIDP    5406    A
COMCIDP  *                                                                       COMCIDP    5407    A
COMCIDP  *         ENTRY  (X5) = FWA OF PARAMETER LIST                           COMCIDP    5408    A
COMCIDP  *                                                                       COMCIDP    5409    A
COMCIDP  *         EXIT   NONE                                                   COMCIDP    5410    A
COMCIDP  *                                                                       COMCIDP    5411    A
COMCIDP  *         USES   ALL BUT A5,X5                                          COMCIDP    5412    A
COMCIDP  *                                                                       COMCIDP    5413    A
COMCIDP  *         CALLS  CDD,FRA,PRIDP                                          COMCIDP    5414    A
COMCIDP                                                                          COMCIDP    5415    A
COMCIDP                                                                          COMCIDP    5416    A
COMCIDP   HDR      SUBR              ** ENTRY/EXIT **                            COMCIDP    5417    A
COMCIDP            SA1    X5+SN=HDR                                              COMCIDP    5418    A
COMCIDP            SA4    X5+SN=CNT                                              COMCIDP    5419    A
COMCIDP            ZR     X1,EXIT.    IF NO HEADER TO BE GENERATED               COMCIDP    5420    A
COMCIDP            MX0    8*CHAR                                                 COMCIDP    5421    A
COMCIDP            BX6    X0*X1                                                  COMCIDP    5422    A
COMCIDP            SX7    2R                                                     COMCIDP    5423    A
COMCIDP            BX6    X6+X7                                                  COMCIDP    5424    A
COMCIDP            LX6    -2*CHAR     (X6) = 10H  NAME                           COMCIDP    5425    A
COMCIDP            SA6    SNAPLNE                                                COMCIDP    5426    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP          SA2    HDRA                                                  COMCIDP    5427    A
COMCIDP          SA3    A2+B1                                                 COMCIDP    5428    A
COMCIDP          SX1    X4                                                    COMCIDP    5429    A
COMCIDP          BX6    X2                                                    COMCIDP    5430    A
COMCIDP          LX7    X3                                                    COMCIDP    5431    A
COMCIDP          SA6    A6+B1                                                 COMCIDP    5432    A
COMCIDP          SA7    A6+2                                                  COMCIDP    5433    A
COMCIDP          RJ     =XCDD       CONVERT BINARY TO DECIMAL DPC             COMCIDP    5434    A
COMCIDP          BX6    X4                                                    COMCIDP    5435    A
COMCIDP          SA6    A6+B1                                                 COMCIDP    5436    A
COMCIDP          SA1    ADR=RJ      (X1) = ADDR OF CALLING RJ                 COMCIDP    5437    A
COMCIDP          RJ     FRA         FIND RELATIVE ADDR                        COMCIDP    5438    A
COMCIDP          PL     B7,HDR2     IF USER PROVIDED *FRA=*                   COMCIDP    5439    A
COMCIDP                                                                       COMCIDP    5440    A
COMCIDP          RJ     =XCOD       CONVERT BINARY TO OCTAL DPC               COMCIDP    5441    A
COMCIDP          BX6    X4                                                    COMCIDP    5442    A
COMCIDP          MX7    0                                                     COMCIDP    5443    A
COMCIDP                                                                       COMCIDP    5444    A
COMCIDP   HDR2   SA6    A7+B1                                                 COMCIDP    5445    A
COMCIDP          SA7    A6+B1                                                 COMCIDP    5446    A
COMCIDP          PRIDP  SNAPLNE,,L.PRB                                        COMCIDP    5447    A
COMCIDP          EQ     EXIT.                                                 COMCIDP    5448    A
COMCIDP                                                                       COMCIDP    5449    A
COMCIDP                                                                       COMCIDP    5450    A
COMCIDP  *       DATA   10H  SNAPNAME                                         COMCIDP    5451    A
COMCIDP   HDRA   DATA   10H SNAP NR                                           COMCIDP    5452    A
COMCIDP  *       DATA   10HNNNNNNN                                            COMCIDP    5453    A
COMCIDP          DATA   10HCALLED BY                                          COMCIDP    5454    A
COMCIDP  *       DATA   10HNNNNNNN IN                                         COMCIDP    5455    A
COMCIDP  *       DATA   10CXXXXXXX                                            COMCIDP    5456    A
COMCIDP   IEX    SPACE  4,10                                                  COMCIDP    5457    A
COMCIDP  **      IEX - INITIALIZE EXECUTIVE.                                  COMCIDP    5458    A
COMCIDP  *                                                                    COMCIDP    5459    A
COMCIDP  *                                                                    COMCIDP    5460    A
COMCIDP  *       ENTRY  (A1,X1) = A+C OF EXECUTIVE ENTRY/EXIT POINT           COMCIDP    5461    A
COMCIDP  *                                                                    COMCIDP    5462    A
COMCIDP  *       EXIT   (B2) = ADDR OF PARAMETER LIST, IF THERE IS ONE.       COMCIDP    5463    A
COMCIDP  *                   = .ZR. IF SPECIAL                                COMCIDP    5464    A
COMCIDP  *                          +  RJ  XEC                                COMCIDP    5465    A
COMCIDP  *                          -  VFD 30/0                               COMCIDP    5466    A
COMCIDP  *                   = .MI. IF NO PARAMETER LIST.                     COMCIDP    5467    A
COMCIDP  *                                                                    COMCIDP    5468    A
COMCIDP  *               (ADR=RJ)  = ADDR OF *RJ XEC* WORD.                   COMCIDP    5469    A
COMCIDP  *               (FW=PARM) = FWA OF PARAMETER LIST FOR THIS EXECUTIVE.COMCIDP    5470    A
COMCIDP  *                           NOTE THAT *IEX* FIGURES OUT WHETHER THERE COMCIDP    5471    A
COMCIDP  *                           REALLY IS A USER-SPECIFIED/ASSEMBLED     COMCIDP    5472    A
COMCIDP  *                           PARAMETER LIST OR NOT.  IF NOT, *IEX* DOES COMCIDP    5473    A
COMCIDP  *                           THE PRELIMINARY INITIALIZATION OF THE DUMMY COMCIDP    5474    A
COMCIDP  *                           PARAMETER LIST AT *APL* ET SEQ.          COMCIDP    5475    A
COMCIDP  *                                                                    COMCIDP    5476    A
COMCIDP  *       USES   ALL                                                   COMCIDP    5477    A
COMCIDP  *                                                                    COMCIDP    5478    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *         CALLS  CXR,SBM                                          COMCIDP     5479     A
COMCIDP                                                                    COMCIDP     5480     A
COMCIDP                                                                    COMCIDP     5481     A
COMCIDP   IEX      SUBR               ** ENTRY/EXIT **                     COMCIDP     5482     A
COMCIDP                                                                    COMCIDP     5483     A
COMCIDP  *         SET UP AND CHECK CALLING *RJ* INSTRUCTION.              COMCIDP     5484     A
COMCIDP                                                                    COMCIDP     5485     A
COMCIDP            LX1    30                                               COMCIDP     5486     A
COMCIDP            SX2    X1-1        (X2) = ADDR OF CALLING *RJ*          COMCIDP     5487     A
COMCIDP            SX1    A1          (X1) = ADDR OF EXECUTIVE             COMCIDP     5488     A
COMCIDP            RJ     CXR         CHECK EXECUTIVE RJ                   COMCIDP     5489     A
COMCIDP            SX6    B2                                               COMCIDP     5490     A
COMCIDP            GT     B2,B0,IEX2  IF PARAMETER LIST LOOKS TO BE PRESENT COMCIDP     5491     A
COMCIDP                                                                    COMCIDP     5492     A
COMCIDP  *         SET UP DUMMY PARAMETER LIST.                            COMCIDP     5493     A
COMCIDP                                                                    COMCIDP     5494     A
COMCIDP            BX6    X6-X6                                            COMCIDP     5495     A
COMCIDP            SX1    L.APL       (X1) = LEN OF DUMMY PARAMETER LIST   COMCIDP     5496     A
COMCIDP            SA6    APL         INITIALIZE (A6)                      COMCIDP     5497     A
COMCIDP            RJ     =XSBM=      SET BLOCK OF MEMORY                  COMCIDP     5498     A
COMCIDP                                                                    COMCIDP     5499     A
COMCIDP  *         SET UP DEFAULT *LL,UL,INC* INTO DUMMY PARAMETER LIST.   COMCIDP     5500     A
COMCIDP                                                                    COMCIDP     5501     A
COMCIDP            SX6    1                                                COMCIDP     5502     A
COMCIDP            SX7    100D                                             COMCIDP     5503     A
COMCIDP            SA6    APL+SN=LL   DEFAULT LL = 1                       COMCIDP     5504     A
COMCIDP            SA7    APL+SN=UL   DEFAULT UL = 100D                    COMCIDP     5505     A
COMCIDP            SA6    APL+SN=INC  DEFAULT INC = 1                      COMCIDP     5506     A
COMCIDP            SX6    APL                                              COMCIDP     5507     A
COMCIDP                                                                    COMCIDP     5508     A
COMCIDP  *         SAVE FWA OF PARAMETER LIST (REAL OR DUMMY).             COMCIDP     5509     A
COMCIDP                                                                    COMCIDP     5510     A
COMCIDP   IEX2     SA6    FW=PARM                                          COMCIDP     5511     A
COMCIDP            EQ     EXIT.                                            COMCIDP     5512     A
COMCIDP   IIF      SPACE  4,10                                             COMCIDP     5513     A
COMCIDP   .IDPOS   IFNE   .OS,2                                            CCGA018       28     A
COMCIDP  **        IIF - INITIALIZE INTERACTIVE FILES.                     COMCIDP     5514     A
COMCIDP  *                                                                 COMCIDP     5515     A
COMCIDP  *                                                                 COMCIDP     5516     A
COMCIDP  *         THIS ROUTINE OPENS AND CONNECTS THE INTERACTIVE DEBUG INPUT COMCIDP 5517     A
COMCIDP  *         AND OUTPUT FILES, *F.IDI* AND *F.IDO*.                  COMCIDP     5518     A
COMCIDP  *                                                                 COMCIDP     5519     A
COMCIDP  *         ENTRY  NONE                                             COMCIDP     5520     A
COMCIDP  *                                                                 COMCIDP     5521     A
COMCIDP  *         EXIT   NONE                                             COMCIDP     5522     A
COMCIDP  *                                                                 COMCIDP     5523     A
COMCIDP  *         USES   ALL BUT A5,X5,A0,X0                              COMCIDP     5524     A
COMCIDP  *                                                                 COMCIDP     5525     A
COMCIDP  *         CALLS  CON,OPEN                                         COMCIDP     5526     A
COMCIDP                                                                    COMCIDP     5527     A
COMCIDP                                                                    COMCIDP     5528     A
COMCIDP   IIF      SUBR               ** ENTRY/EXIT **                     COMCIDP     5529     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP                                                                    COMCIDP     5530     A
COMCIDP  *         SET UP *F.IDI - INTERACTIVE DEBUG INPUT FILE*.          COMCIDP     5531     A
COMCIDP                                                                    COMCIDP     5532     A
COMCIDP            SA1    IDPFLG                                           COMCIDP     5533     A
COMCIDP            MX2    1                                                COMCIDP     5534     A
COMCIDP            LX2    IDF.RAPP-59                                      COMCIDP     5535     A
COMCIDP            BX6    -X2*X1      CLEAR *RAP*                          COMCIDP     5536     A
COMCIDP            LX2    IDF.INPP-IDF.RAPP                                COMCIDP     5537     A
COMCIDP            BX6    -X2*X6      CLEAR *INP*                          COMCIDP     5538     A
COMCIDP            SA6    A1                                               COMCIDP     5539     A
COMCIDP            OPEN   F.IDI,,RCL                                       COMCIDP     5540     A
COMCIDP            SA1    F.IDI                                            COMCIDP     5541     A
COMCIDP            MX2    0           SET TO *CONNECT*                     COMCIDP     5542     A
COMCIDP            RJ     CON         CONNECT F.IDI                        COMCIDP     5543     A
COMCIDP                                                                    COMCIDP     5544     A
COMCIDP   #OS3     IFEQ   .OS,3       IF SCOPE 3                           COMCIDP     5545     A
COMCIDP            MI     X1,*+4S15   IF *CON* DETECTED ERROR...           COMCIDP     5546     A
COMCIDP   #OS3     ENDIF                                                   COMCIDP     5547     A
COMCIDP                                                                    COMCIDP     5548     A
COMCIDP  *         SET UP *F.IDO - INTERACTIVE DEBUG OUTPUT FILE*.         COMCIDP     5549     A
COMCIDP                                                                    COMCIDP     5550     A
COMCIDP            OPEN   F.IDO,,RCL                                       COMCIDP     5551     A
COMCIDP            SA1    F.IDO                                            COMCIDP     5552     A
COMCIDP            MX2    0           SET TO *CONNECT*                     COMCIDP     5553     A
COMCIDP            RJ     CON         CONNECT F.IDO                        COMCIDP     5554     A
COMCIDP                                                                    COMCIDP     5555     A
COMCIDP   #OS3     IFEQ   .OS,3       IF SCOPE 3                           COMCIDP     5556     A
COMCIDP            MI     X1,*+4S15   IF *CON* DETECTED ERROR...           COMCIDP     5557     A
COMCIDP   #OS3     ENDIF                                                   COMCIDP     5558     A
COMCIDP                                                                    COMCIDP     5559     A
COMCIDP            EQ     EXIT.                                            COMCIDP     5560     A
COMCIDP   IST      SPACE  4,10                                             COMCIDP     5561     A
COMCIDP  **        IST - INITIALIZE *SET* TABLE.                           COMCIDP     5562     A
COMCIDP  *                                                                 COMCIDP     5563     A
COMCIDP  *                                                                 COMCIDP     5564     A
COMCIDP  *                THIS ROUTINE SETS UP THE DEFAULT *SET* TABLE VALUES.  COMCIDP  5565  A
COMCIDP  *                                                                 COMCIDP     5566     A
COMCIDP  *         ENTRY  NONE                                             COMCIDP     5567     A
COMCIDP  *                                                                 COMCIDP     5568     A
COMCIDP  *         EXIT   NONE                                             COMCIDP     5569     A
COMCIDP  *                                                                 COMCIDP     5570     A
COMCIDP  *         USES   ALL                                              COMCIDP     5571     A
COMCIDP  *                                                                 COMCIDP     5572     A
COMCIDP  *         CALLS  ADZ                                              COMCIDP     5573     A
COMCIDP                                                                    COMCIDP     5574     A
COMCIDP                                                                    COMCIDP     5575     A
COMCIDP   IST      SUBR               ** ENTRY/EXIT **                     COMCIDP     5576     A
COMCIDP            SB6    IDPSET      (B6) = FWA OF IDP *SET* TABLE        COMCIDP     5577     A
COMCIDP            SA1    =4LTEMP                                          COMCIDP     5578     A
COMCIDP            SX6    IDPTMP                                           COMCIDP     5579     A
COMCIDP            BX6    X1+X6       (X6) = 42/0LTEMP,18/FWA OF TEMP AREA COMCIDP     5580     A
COMCIDP            RJ     ADZ         ADD WORD TO IDP TABLE                COMCIDP     5581     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            SX7    1RP                                                    COMCIDP     5582     A
COMCIDP            SX6    IDPPREG                                                COMCIDP     5583     A
COMCIDP            LX7    10*CHAR-1*CHAR                                         COMCIDP     5584     A
COMCIDP            BX6    X7+X6       (X6) = 42/0LP, 18/ADDR OF PSEUDO P REGISTER COMCIDP     5585     A
COMCIDP            RJ     ADZ         ADD WORD TO IDP TABLE                      COMCIDP     5586     A
COMCIDP            EQ     EXIT.                                                  COMCIDP     5587     A
COMCIDP   LBT      SPACE  4,10                                                   COMCIDP     5588     A
COMCIDP  **        LBT - LIST BREAK TABLE.                                       COMCIDP     5589     A
COMCIDP  *                                                                       COMCIDP     5590     A
COMCIDP  *                                                                       COMCIDP     5591     A
COMCIDP  *                LISTS THE CONTENTS OF THE PARALLEL BREAK TABLES        COMCIDP     5592     A
COMCIDP  *         *IDPBA/IDPBC* IN THREE FORMS, DEPENDING ON THE TYPE OF BREAK-- COMCIDP     5593     A
COMCIDP  *                                                                       COMCIDP     5594     A
COMCIDP  *     COL 1         1         1         1        (.=BLANK(55B))        COMCIDP     5595     A
COMCIDP  *         .BREAK.AT.NNNNNN.IN.XXXXXXX                                   COMCIDP     5596     A
COMCIDP  *         ...........LL.NNNNN..UL.NNNNN..INC.NNNN.CNT.NNNNNN            COMCIDP     5597     A
COMCIDP  *                                                                       COMCIDP     5598     A
COMCIDP  *         .BRPL..AT.NNNNNN.IN.XXXXXXX....A1.NNNNNN                      COMCIDP     5599     A
COMCIDP  *         ...........LL.NNNNN..UL.NNNNN..INC.NNNN.CNT.NNNNNN            COMCIDP     5600     A
COMCIDP  *                                                                       COMCIDP     5601     A
COMCIDP  *         .BREQ..AT.NNNNNN.IN.XXXXXXX....A1.NNNNNN.A2.NNNNNN            COMCIDP     5602     A
COMCIDP  *         ...........LL.NNNNN..UL.NNNNN..INC.NNNN.CNT.NNNNNN            COMCIDP     5603     A
COMCIDP  *                                                                       COMCIDP     5604     A
COMCIDP  *         ENTRY  NONE                                                   COMCIDP     5605     A
COMCIDP  *                                                                       COMCIDP     5606     A
COMCIDP  *         EXIT   NONE                                                   COMCIDP     5607     A
COMCIDP  *                                                                       COMCIDP     5608     A
COMCIDP  *         USES   ALL BUT A0                                             COMCIDP     5609     A
COMCIDP  *                                                                       COMCIDP     5610     A
COMCIDP  *         CALLS  COD,FRA,PRIDP,SFN                                      COMCIDP     5611     A
COMCIDP                                                                          COMCIDP     5612     A
COMCIDP                                                                          COMCIDP     5613     A
COMCIDP   LBT      SUBR               ** ENTRY/EXIT **                           COMCIDP     5614     A
COMCIDP                                                                          COMCIDP     5615     A
COMCIDP  *         SET UP FOR 1ST ENTRY.                                         COMCIDP     5616     A
COMCIDP                                                                          COMCIDP     5617     A
COMCIDP            SA5    IDPBA                                                  COMCIDP     5618     A
COMCIDP            EQ     LBT2                                                   COMCIDP     5619     A
COMCIDP                                                                          COMCIDP     5620     A
COMCIDP  *         MAIN LOOP.                                                    COMCIDP     5621     A
COMCIDP                                                                          COMCIDP     5622     A
COMCIDP   LBT1     SA5    A5+B1                                                  COMCIDP     5623     A
COMCIDP                                                                          COMCIDP     5624     A
COMCIDP   LBT2     ZR     X5,EXIT.    IF FINISHED LISTING ALL BREAKS             COMCIDP     5625     A
COMCIDP            SX6    X5+B1                                                  COMCIDP     5626     A
COMCIDP            ZR     X6,LBT1     IF THIS ENTRY AVAILABLE (NOT A BREAK)      COMCIDP     5627     A
COMCIDP                                                                          COMCIDP     5628     A
COMCIDP  *         ASSEMBLE * BREAK AT *.                                        COMCIDP     5629     A
COMCIDP                                                                          COMCIDP     5630     A
COMCIDP            SX4    X5          (X4) = BREAK ADDRESS                       COMCIDP     5631     A
COMCIDP            LX5    6                                                      COMCIDP     5632     A
COMCIDP            MX0    -6                                                     COMCIDP     5633     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP          BX6    -X0*X5      (X6) = BREAK TYPE CODE                    COMCIDP    5634    A
COMCIDP          MX0    7*CHAR                                                COMCIDP    5635    A
COMCIDP          IX2    X6+X6       *2                                        COMCIDP    5636    A
COMCIDP          SA1    ST.BRK+X2   (X1) = *KEYW* ENTRY FOR THIS BREAK        COMCIDP    5637    A
COMCIDP          SA6    IDPBTC                                                COMCIDP    5638    A
COMCIDP          LX1    0-KW.KEYP                                             COMCIDP    5639    A
COMCIDP          SA1    X1          (X1) = KEYW LITERAL FOR THIS BREAK, -L- FMT COMCIDP  5640    A
COMCIDP          RJ     =XSFN       SPACE FILL NAME                           COMCIDP    5641    A
COMCIDP          LX6    -CHAR                                                 COMCIDP    5642    A
COMCIDP          BX6    X0*X6                                                 COMCIDP    5643    A
COMCIDP          SX7    3RAT                                                  COMCIDP    5644    A
COMCIDP          BX6    X6+X7       (X6) = .BREAK.AT.   (.=BLANK(55B))        COMCIDP    5645    A
COMCIDP          SA6    SNAPLNE                                               COMCIDP    5646    A
COMCIDP                                                                       COMCIDP    5647    A
COMCIDP  *       ASSEMBLE *NNNNNN IN XXXXXXX   *.                              COMCIDP    5648    A
COMCIDP                                                                       COMCIDP    5649    A
COMCIDP          SX1    X4                                                    COMCIDP    5650    A
COMCIDP          RJ     FRA         FIND RELATIVE ADDRESS                     COMCIDP    5651    A
COMCIDP          PL     B7,LBT3     IF USER PROVIDED *FRA=*                   COMCIDP    5652    A
COMCIDP                                                                       COMCIDP    5653    A
COMCIDP          RJ     =XCOD       CONVERT BINARY TO OCTAL DPC               COMCIDP    5654    A
COMCIDP          BX6    X4                                                    COMCIDP    5655    A
COMCIDP          MX7    0                                                     COMCIDP    5656    A
COMCIDP                                                                       COMCIDP    5657    A
COMCIDP   LBT3   SA6    A6+B1                                                 COMCIDP    5658    A
COMCIDP          SA7    A6+B1                                                 COMCIDP    5659    A
COMCIDP          SA2    IDPBTC                                                COMCIDP    5660    A
COMCIDP          ZR     X2,LBT5     IF *BREAK ADDR,LL,UL,INC* FORM            COMCIDP    5661    A
COMCIDP          BX1    X7                                                    COMCIDP    5662    A
COMCIDP          RJ     SFN         SPACE FILL NAME                           COMCIDP    5663    A
COMCIDP          SA6    A6+B1       RE-STORE BLANK FILLED *XXXXXXX   *        COMCIDP    5664    A
COMCIDP                                                                       COMCIDP    5665    A
COMCIDP  *       ASSEMBLE * A1 NNNNNN*.                                        COMCIDP    5666    A
COMCIDP                                                                       COMCIDP    5667    A
COMCIDP          LX5    18                                                    COMCIDP    5668    A
COMCIDP          SX1    X5                                                    COMCIDP    5669    A
COMCIDP          RJ     =XCOD       CONVERT BINARY TO OCTAL DPC               COMCIDP    5670    A
COMCIDP          SA2    IDPBTC                                                COMCIDP    5671    A
COMCIDP          SX1    3R A1                                                 COMCIDP    5672    A
COMCIDP          MX0    -7*CHAR                                               COMCIDP    5673    A
COMCIDP          BX6    -X0*X6                                                COMCIDP    5674    A
COMCIDP          LX1    10*CHAR-3*CHAR                                        COMCIDP    5675    A
COMCIDP          BX1    X0*X1                                                 COMCIDP    5676    A
COMCIDP          BX6    X6+X1       (X6) = .A1.NNNNNN   (.=BLANK(55B))        COMCIDP    5677    A
COMCIDP          SA6    A6+B1                                                 COMCIDP    5678    A
COMCIDP          SX2    X2-BTC.EQ                                             COMCIDP    5679    A
COMCIDP          MI     X2,LBT4     IF *BRPL ADDR,A1,LL,UL,INC* FORM          COMCIDP    5680    A
COMCIDP                                                                       COMCIDP    5681    A
COMCIDP  *       ASSEMBLE * A2 NNNNNN*.                                        COMCIDP    5682    A
COMCIDP                                                                       COMCIDP    5683    A
COMCIDP          LX5    18                                                    COMCIDP    5684    A
COMCIDP          SX1    X5                                                    COMCIDP    5685    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            RJ     =XCOD       CONVERT BINARY TO OCTAL DPC                    COMCIDP     5686     A
COMCIDP            SX1    3R A2                                                      COMCIDP     5687     A
COMCIDP            MX0    -7*CHAR                                                    COMCIDP     5688     A
COMCIDP            BX6    -X0*X6                                                     COMCIDP     5689     A
COMCIDP            LX1    10*CHAR-3*CHAR                                             COMCIDP     5690     A
COMCIDP            BX1    X0*X1                                                      COMCIDP     5691     A
COMCIDP            BX6    X6+X1       (X6) = .A2.NNNNNN   (.=BLANK(55B))             COMCIDP     5692     A
COMCIDP            SA6    A6+B1                                                      COMCIDP     5693     A
COMCIDP                                                                              COMCIDP     5694     A
COMCIDP   LBT4     MX6    0                                                          COMCIDP     5695     A
COMCIDP            SA6    A6+B1       MARK EOL                                       COMCIDP     5696     A
COMCIDP                                                                              COMCIDP     5697     A
COMCIDP   LBT5     PRIDP  SNAPLNE                                                    COMCIDP     5698     A
COMCIDP                                                                              COMCIDP     5699     A
COMCIDP   *        ASSEMBLE *           LL NNNNN *.                                 COMCIDP     5700     A
COMCIDP                                                                              COMCIDP     5701     A
COMCIDP            SA1    =10H                                                       COMCIDP     5702     A
COMCIDP            SB7    A5-IDPBA                                                   COMCIDP     5703     A
COMCIDP            SB6    B7+B7                                                      COMCIDP     5704     A
COMCIDP            SA2    IDPBC+B6+1  (X2) = 2ND WORD OF *IDPBC* ENTRY               COMCIDP     5705     A
COMCIDP            BX6    X1                                                         COMCIDP     5706     A
COMCIDP            BX5    X2                                                         COMCIDP     5707     A
COMCIDP            LX5    15                                                         COMCIDP     5708     A
COMCIDP            MX0    -15                                                        COMCIDP     5709     A
COMCIDP            BX1    -X0*X5                                                     COMCIDP     5710     A
COMCIDP            SA6    SNAPLNE                                                    COMCIDP     5711     A
COMCIDP            RJ     =XCOD       CONVERT BINARY TO OCTAL DPC                    COMCIDP     5712     A
COMCIDP            SX1    3R LL                                                      COMCIDP     5713     A
COMCIDP            LX4    -4*CHAR     LEFT JUSTIFY NR TO BIT 35                      COMCIDP     5714     A
COMCIDP            BX6    X4          (X6) = ....NNNNN.   (.=BLANK(55B))             COMCIDP     5715     A
COMCIDP            MX0    -7*CHAR                                                    COMCIDP     5716     A
COMCIDP            LX1    10*CHAR-3*CHAR                                             COMCIDP     5717     A
COMCIDP            BX1    X0*X1                                                      COMCIDP     5718     A
COMCIDP            BX6    -X0*X6                                                     COMCIDP     5719     A
COMCIDP            BX6    X6+X1       (X6) = .LL.NNNNN.   (.=BLANK(55B))             COMCIDP     5720     A
COMCIDP            SA6    A6+B1                                                      COMCIDP     5721     A
COMCIDP                                                                              COMCIDP     5722     A
COMCIDP   *        ASSEMBLE * UL NNNNN *.                                            COMCIDP     5723     A
COMCIDP                                                                              COMCIDP     5724     A
COMCIDP            LX5    15                                                         COMCIDP     5725     A
COMCIDP            MX0    -15                                                        COMCIDP     5726     A
COMCIDP            BX1    -X0*X5      (X1) = UL                                      COMCIDP     5727     A
COMCIDP            RJ     =XCOD       CONVERT BINARY TO OCTAL DPC                    COMCIDP     5728     A
COMCIDP            SX1    3R UL                                                      COMCIDP     5729     A
COMCIDP            LX4    -4*CHAR                                                    COMCIDP     5730     A
COMCIDP            BX6    X4          (X6) = ....NNNNN.   (.=BLANK(55B))             COMCIDP     5731     A
COMCIDP            MX0    -7*CHAR                                                    COMCIDP     5732     A
COMCIDP            BX6    -X0*X6                                                     COMCIDP     5733     A
COMCIDP            LX1    10*CHAR-3*CHAR                                             COMCIDP     5734     A
COMCIDP            BX1    X0*X1                                                      COMCIDP     5735     A
COMCIDP            BX6    X6+X1       (X6) = .UL.NNNNN.   (.=BLANK(55B))             COMCIDP     5736     A
COMCIDP            SA6    A6+B1                                                      COMCIDP     5737     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP                                                                                COMCIDP     5738     A
COMCIDP  *         ASSEMBLE * INC NNNN *.                                              COMCIDP     5739     A
COMCIDP                                                                                COMCIDP     5740     A
COMCIDP            LX5    12                                                           COMCIDP     5741     A
COMCIDP            MX0    -12                                                          COMCIDP     5742     A
COMCIDP            BX1    -X0*X5      (X1) = INC                                       COMCIDP     5743     A
COMCIDP            RJ     =XCOD       CONVERT BINARY TO OCTAL DPC                      COMCIDP     5744     A
COMCIDP            SA1    =4L INC                                                      COMCIDP     5745     A
COMCIDP            LX4    -5*CHAR                                                      COMCIDP     5746     A
COMCIDP            BX6    X4          (X6) = .....NNNN.   (.=BLANK(55B))               COMCIDP     5747     A
COMCIDP            MX0    -6*CHAR                                                      COMCIDP     5748     A
COMCIDP            BX6    -X0*X6                                                       COMCIDP     5749     A
COMCIDP            BX6    X6+X1       (X6) = .INC.NNNN.   (.=BLANK(55B))               COMCIDP     5750     A
COMCIDP            SA6    A6+B1                                                        COMCIDP     5751     A
COMCIDP                                                                                COMCIDP     5752     A
COMCIDP  *         ASSEMBLE *CNT NNNNNN*.                                              COMCIDP     5753     A
COMCIDP                                                                                COMCIDP     5754     A
COMCIDP            LX5    18                                                           COMCIDP     5755     A
COMCIDP            SX1    X5                                                           COMCIDP     5756     A
COMCIDP            RJ     =XCOD       CONVERT BINARY TO OCTAL DPC                      COMCIDP     5757     A
COMCIDP            SX1    3RCNT                                                        COMCIDP     5758     A
COMCIDP            MX0    -7*CHAR                                                      COMCIDP     5759     A
COMCIDP            BX6    -X0*X6                                                       COMCIDP     5760     A
COMCIDP            LX1    10*CHAR-3*CHAR                                               COMCIDP     5761     A
COMCIDP            BX1    X0*X1                                                        COMCIDP     5762     A
COMCIDP            BX6    X6+X1       (X6) = CNT.NNNNNN   (.=BLANK(55B))               COMCIDP     5763     A
COMCIDP            MX7    0                                                            COMCIDP     5764     A
COMCIDP            SA6    A6+B1                                                        COMCIDP     5765     A
COMCIDP            SA7    A6+B1       MARK EOL                                         COMCIDP     5766     A
COMCIDP            PRIDP  SNAPLNE                                                      COMCIDP     5767     A
COMCIDP            EQ     LBT1                                                         COMCIDP     5768     A
COMCIDP   LST      SPACE  4,10                                                         COMCIDP     5769     A
COMCIDP  **        LST - LIST *SET* NAME TABLE.                                        COMCIDP     5770     A
COMCIDP  *                                                                             COMCIDP     5771     A
COMCIDP  *                                                                             COMCIDP     5772     A
COMCIDP  *         ENTRY  (A5,X5) =   ADDR + CNTS OF 1ST *SET* TABLE ENTRY             COMCIDP     5773     A
COMCIDP  *                                                                             COMCIDP     5774     A
COMCIDP  *         EXIT   NONE                                                         COMCIDP     5775     A
COMCIDP  *                                                                             COMCIDP     5776     A
COMCIDP  *         USES   ALL                                                          COMCIDP     5777     A
COMCIDP  *                                                                             COMCIDP     5778     A
COMCIDP  *         CALLS  COD,FRA,PRIDP,SFN                                            COMCIDP     5779     A
COMCIDP                                                                                COMCIDP     5780     A
COMCIDP                                                                                COMCIDP     5781     A
COMCIDP   LST      SUBR               ** ENTRY/EXIT **                                 COMCIDP     5782     A
COMCIDP                                                                                COMCIDP     5783     A
COMCIDP   LST2     ZR     X5,EXIT.    IF FINISHED ALL SET NAMES                        COMCIDP     5784     A
COMCIDP            SX1    B1                                                           COMCIDP     5785     A
COMCIDP            IX6    X5+X1                                                        COMCIDP     5786     A
COMCIDP            ZR     X6,LST4     IF THIS ENTRY IS AVAILABLE (NOT A SET)           COMCIDP     5787     A
COMCIDP            MX0    7*CHAR                                                       COMCIDP     5788     A
COMCIDP            BX1    X0*X5       (X1) = 0LNAME                                    COMCIDP     5789     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            RJ     =XSFN       SPACE FILL NAME                                COMCIDP    5790    A
COMCIDP            LX6    -CHAR                                                      COMCIDP    5791    A
COMCIDP            SA6    SNAPLNE                                                    COMCIDP    5792    A
COMCIDP            BX1    -X0*X5                                                     COMCIDP    5793    A
COMCIDP            RJ     =XCOD       CONVERT BINARY TO OCTAL DPC-- SET NAME VALU    COMCIDP    5794    A
COMCIDP            LX6    4*CHAR                                                     COMCIDP    5795    A
COMCIDP            SA6    A6+B1                                                      COMCIDP    5796    A
COMCIDP            BX1    -X0*X5                                                     COMCIDP    5797    A
COMCIDP            RJ     FRA         FIND RELATIVE ADDRESS                          COMCIDP    5798    A
COMCIDP            SA6    A6+B1                                                      COMCIDP    5799    A
COMCIDP            SA7    A6+B1                                                      COMCIDP    5800    A
COMCIDP            PRIDP  SNAPLNE                                                    COMCIDP    5801    A
COMCIDP                                                                              COMCIDP    5802    A
COMCIDP   LST4     SA5    A5+B1                                                      COMCIDP    5803    A
COMCIDP            EQ     LST2                                                       COMCIDP    5804    A
COMCIDP   LXT      SPACE  4,10                                                       COMCIDP    5805    A
COMCIDP  **        LXT - LIST XFER TABLE.                                            COMCIDP    5806    A
COMCIDP  *                                                                           COMCIDP    5807    A
COMCIDP  *                                                                           COMCIDP    5808    A
COMCIDP  *                LISTS THE MOST RECENT TRANSFERS OF CONTROL TO OCCUR        COMCIDP    5809    A
COMCIDP  *         DURING STEP MODE (CONTAINED IN *IDPXFT*).                         COMCIDP    5810    A
COMCIDP  *                                                                           COMCIDP    5811    A
COMCIDP  *         *IDPXFT* FORMAT --                                                COMCIDP    5812    A
COMCIDP  *                                                                           COMCIDP    5813    A
COMCIDP  *         VFD    60/-1       UNUSED ENTRY (I.E. XFT HAS NOT FILLED YET)     COMCIDP    5814    A
COMCIDP  *          .                                                                COMCIDP    5815    A
COMCIDP  *          .                                                                COMCIDP    5816    A
COMCIDP  *         VFD    30/POS COUNTER,30/P REGISTER OF XFER INSTRUCTION           COMCIDP    5817    A
COMCIDP  *          .                                                                COMCIDP    5818    A
COMCIDP  *         VFD    60/0        END OF TABLE MARK                              COMCIDP    5819    A
COMCIDP  *                                                                           COMCIDP    5820    A
COMCIDP  *         ENTRY  (X6) = NR OF ENTRIES TO LIST (BEGINNING W/ MOST RECENT)    COMCIDP    5821    A
COMCIDP  *                                                                           COMCIDP    5822    A
COMCIDP  *         EXIT   NONE                                                       COMCIDP    5823    A
COMCIDP  *                                                                           COMCIDP    5824    A
COMCIDP  *         USES   ALL                                                        COMCIDP    5825    A
COMCIDP  *                                                                           COMCIDP    5826    A
COMCIDP  *         CALLS  DAZ                                                        COMCIDP    5827    A
COMCIDP                                                                              COMCIDP    5828    A
COMCIDP                                                                              COMCIDP    5829    A
COMCIDP   LXT      SUBR               ** ENTRY/EXIT **                               COMCIDP    5830    A
COMCIDP                                                                              COMCIDP    5831    A
COMCIDP  *         INITIALIZATION.                                                   COMCIDP    5832    A
COMCIDP                                                                              COMCIDP    5833    A
COMCIDP            SX1    IDPXFT+L.XFT                                               COMCIDP    5834    A
COMCIDP            SX7    X6-L.XFT-1                                                 COMCIDP    5835    A
COMCIDP            MI     X7,LXT2     IF NOT ASKING FOR TOO MANY ENTRIES             COMCIDP    5836    A
COMCIDP            SX6    L.XFT                                                      COMCIDP    5837    A
COMCIDP                                                                              COMCIDP    5838    A
COMCIDP   LXT2     IX7    X1-X6                                                      COMCIDP    5839    A
COMCIDP            NO                                                                COMCIDP    5840    A
COMCIDP            SA7    LXT=NXT                                                    COMCIDP    5841    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP                                                                  COMCIDP    5842    A
COMCIDP  *         ** MAIN LOOP NODE **                                  COMCIDP    5843    A
COMCIDP                                                                  COMCIDP    5844    A
COMCIDP   LXT3     SA2    LXT=NXT                                        COMCIDP    5845    A
COMCIDP            SA3    X2          (X3) = NEXT *XFT* ENTRY            COMCIDP    5846    A
COMCIDP            SX7    X2+B1                                          COMCIDP    5847    A
COMCIDP            SX4    X3+B1                                          COMCIDP    5848    A
COMCIDP            SA7    A2                                             COMCIDP    5849    A
COMCIDP            ZR     X3,EXIT.    IF HIT END OF TABLE MARK, DONE...  COMCIDP    5850    A
COMCIDP            ZR     X4,LXT3     IF HIT AVAILABLE ENTRY (I.E. *XFT* WAS NOT  COMCIDP    5851    A
COMCIDP  *                               FULL TO BEGIN WITH)             COMCIDP    5852    A
COMCIDP            SA5    X3          (A5,X5) = A+C OF XFER INSTRUCTION  COMCIDP    5853    A
COMCIDP            AX3    30                                             COMCIDP    5854    A
COMCIDP            SB4    X3          (B4) = POS COUNTER                 COMCIDP    5855    A
COMCIDP            SA7    IDPFAD      SET TO *FORCE OUT ADDR UNCONDITIONALLY*  COMCIDP    5856    A
COMCIDP            RJ     DAZ         DEASSEMBLE XFER INSTRUCTION        COMCIDP    5857    A
COMCIDP            EQ     LXT3                                           COMCIDP    5858    A
COMCIDP                                                                  COMCIDP    5859    A
COMCIDP                                                                  COMCIDP    5860    A
COMCIDP   LXT=NXT  =      APL         SAVED ADDR OF NEXT *XFT* ENTRY     COMCIDP    5861    A
COMCIDP   MUL      SPACE  4,10                                           COMCIDP    5862    A
COMCIDP  **        MUL - INTEGER MULTIPLY.                               COMCIDP    5863    A
COMCIDP  *                                                               COMCIDP    5864    A
COMCIDP  *                                                               COMCIDP    5865    A
COMCIDP  *         *MUL* INTEGER MULTIPLIES THE 2 **POSITIVE** INTEGERS  COMCIDP    5866    A
COMCIDP  *         IN REGISTERS (X1) AND (X2), PRODUCING A 60 BIT RESULT.  COMCIDP    5867    A
COMCIDP  *                                                               COMCIDP    5868    A
COMCIDP  *         *MUL* IS ONLY NECESSARY BECAUSE THE HARDWARE CAN ONLY GET  COMCIDP    5869    A
COMCIDP  *         A MAXIMUM OF 48 BITS OF PRECISION ON AN INTEGER MULTIPLY, AND  COMCIDP    5870    A
COMCIDP  *         WE NEED 60 BITS.                                      COMCIDP    5871    A
COMCIDP  *                                                               COMCIDP    5872    A
COMCIDP  *         ENTRY  (X1) = X                                       COMCIDP    5873    A
COMCIDP  *                (X2) = Y                                       COMCIDP    5874    A
COMCIDP  *                                                               COMCIDP    5875    A
COMCIDP  *         EXIT   (X7) = X*Y  (60 BIT RESULT).                   COMCIDP    5876    A
COMCIDP  *                                                               COMCIDP    5877    A
COMCIDP  *         USES   X - 1,2,3,4,7                                  COMCIDP    5878    A
COMCIDP  *                A - NONE                                       COMCIDP    5879    A
COMCIDP  *                B - NONE                                       COMCIDP    5880    A
COMCIDP  *                                                               COMCIDP    5881    A
COMCIDP  *         CALLS  NONE                                           COMCIDP    5882    A
COMCIDP                                                                  COMCIDP    5883    A
COMCIDP                                                                  COMCIDP    5884    A
COMCIDP   MUL      SUBR               ** ENTRY/EXIT **                   COMCIDP    5885    A
COMCIDP            MI     X1,*+4S15   IF X IS NEGATIVE, CAN'T DO IT...   COMCIDP    5886    A
COMCIDP            MI     X2,*+4S15   IF Y IS NEGATIVE, CAN'T DO IT...   COMCIDP    5887    A
COMCIDP            SX3    1                                              COMCIDP    5888    A
COMCIDP            BX7    X7-X7                                          COMCIDP    5889    A
COMCIDP            IX4    X1-X2                                          COMCIDP    5890    A
COMCIDP            PL     X4,MUL2     IF X .GT. Y                        COMCIDP    5891    A
COMCIDP            BX1    X1-X2                                          COMCIDP    5892    A
COMCIDP            BX2    X1-X2                                          COMCIDP    5893    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            BX1    X1-X2                                                  COMCIDP     5894     A
COMCIDP                                                                           COMCIDP     5895     A
COMCIDP   MUL2     ZR     X2,EXIT.    IF DONE                                    COMCIDP     5896     A
COMCIDP            IX7    X7+X1                                                  COMCIDP     5897     A
COMCIDP            IX2    X2-X3                                                  COMCIDP     5898     A
COMCIDP            EQ     MUL2                                                   COMCIDP     5899     A
COMCIDP   PAS      SPACE  4,10                                                   COMCIDP     5900     A
COMCIDP  **        PAS - PARSE SUBEXPRESSION.                                    COMCIDP     5901     A
COMCIDP  *                                                                       COMCIDP     5902     A
COMCIDP  *                                                                       COMCIDP     5903     A
COMCIDP  *         THIS IS THE PARSER...                                         COMCIDP     5904     A
COMCIDP  *                                                                       COMCIDP     5905     A
COMCIDP  *         *PAS* IS A RELATIVELY SIMPLE 2 PASS REVERSE POLISH NOTATION   COMCIDP     5906     A
COMCIDP  *         (*RPN*) PARSER.  IT IS USED BY JUST ABOUT EVERYONE IN IDP FOR COMCIDP     5907     A
COMCIDP  *         EVALUATING EXPRESSIONS TYPED IN BY FRIENDLY HUMANS.           COMCIDP     5908     A
COMCIDP  *                                                                       COMCIDP     5909     A
COMCIDP  *         *PAS* CAN BE SPLIT DOWN THE MIDDLE INTO 2 PHASES: POLISH OUT  COMCIDP     5910     A
COMCIDP  *         (I.E. THE GENERATION OF REVERSE POLISH NOTATION TO *RPN*      COMCIDP     5911     A
COMCIDP  *         TABLE), AND POLISH IN (I.E. THE EVALUATION OF THE *RPN* THAT  COMCIDP     5912     A
COMCIDP  *         WE JUST GENERATED INTO A SINGLE BINARY ANSWER/VALUE).         COMCIDP     5913     A
COMCIDP  *                                                                       COMCIDP     5914     A
COMCIDP  *         FOLLOWING IS A PICTORIAL OF *DATA FLOW* IN *PAS* --           COMCIDP     5915     A
COMCIDP  *                                                                       COMCIDP     5916     A
COMCIDP  *                                                                       COMCIDP     5917     A
COMCIDP  *                    POLISH OUT      I     POLISH IN                    COMCIDP     5918     A
COMCIDP  *                                    I                                  COMCIDP     5919     A
COMCIDP  *                       +------+     I                                  COMCIDP     5920     A
COMCIDP  *                       I      I     I                                  COMCIDP     5921     A
COMCIDP  *                       I PAST I     I                                  COMCIDP     5922     A
COMCIDP  *                       I      I     I                                  COMCIDP     5923     A
COMCIDP  *                       +------+     I                                  COMCIDP     5924     A
COMCIDP  *                         &  .       I                                  COMCIDP     5925     A
COMCIDP  *               +----+    .  .    +-----+      +------+                COMCIDP     5926     A
COMCIDP  *               I    I 1  .  .    I     I      I      I                 COMCIDP     5927     A
COMCIDP  *               I    I ....  ...> I     I 2    I      I                 COMCIDP     5928     A
COMCIDP  *               I TB I           I RPN I ...> I PAST I ...> (IDPACC)    COMCIDP     5929     A
COMCIDP  *               I    I .........> I     I      I      I                 COMCIDP     5930     A
COMCIDP  *               I    I 2          I     I      I      I                 COMCIDP     5931     A
COMCIDP  *               +----+            +-----+      +------+                COMCIDP     5932     A
COMCIDP  *                                    I                                  COMCIDP     5933     A
COMCIDP  *                                    I                                  COMCIDP     5934     A
COMCIDP  *                                                                       COMCIDP     5935     A
COMCIDP  *         WHERE  TB   = TOKEN BUFFER (I.E. *PAS* INPUT).                COMCIDP     5936     A
COMCIDP  *                RPN  = REVERSE POLISH NOTATION TABLE (I.E. POLISH      COMCIDP     5937     A
COMCIDP  *                       OUT OUTPUT, AND POLISH IN INPUT).               COMCIDP     5938     A
COMCIDP  *                PAST = PARSING STACK (USED DURING POLISH OUT FOR       COMCIDP     5939     A
COMCIDP  *                       ORDERING OPERATORS, AND DURING POLISH IN        COMCIDP     5940     A
COMCIDP  *                       FOR EVALUATING OPERANDS).                       COMCIDP     5941     A
COMCIDP  *                                                                       COMCIDP     5942     A
COMCIDP  *                1    = DATA FLOW PATH FOR OPERATORS.                   COMCIDP     5943     A
COMCIDP  *                2    = DATA FLOW PATH FOR OPERANDS.                    COMCIDP     5944     A
COMCIDP  *                                                                       COMCIDP     5945     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *                IDPACC = THE ACCUMULATOR.  I.E. WHERE *PAS* STUFFS THE          COMCIDP     5946     A
COMCIDP  *                       FINAL RESULT RIGHT BEFORE EXITING.                       COMCIDP     5947     A
COMCIDP  *                                                                                 COMCIDP     5948     A
COMCIDP  *                                                                                 COMCIDP     5949     A
COMCIDP  *         THE GENERAL ALGORITHM USED IN THIS PARSER CAN BE REPRESENTED            COMCIDP     5950     A
COMCIDP  *         IN A PSEUDO ENGLISH/PASCAL AS FOLLOWS --                                COMCIDP     5951     A
COMCIDP  *                                                                                 COMCIDP     5952     A
COMCIDP  *         POLISH OUT --                                                           COMCIDP     5953     A
COMCIDP  *                                                                                 COMCIDP     5954     A
COMCIDP  *                CASE OF TOKEN TYPE                                               COMCIDP     5955     A
COMCIDP  *                                                                                 COMCIDP     5956     A
COMCIDP  *                   OPERAND:  PASS TO *RPN*.                                      COMCIDP     5957     A
COMCIDP  *                                                                                 COMCIDP     5958     A
COMCIDP  *                   OPERATOR: WHILE PRIORITY OF TOKEN .LE. PRIORITY               COMCIDP     5959     A
COMCIDP  *                             OF TOP-OF-STACK, PASS TOP-OF-STACK                  COMCIDP     5960     A
COMCIDP  *                             TO *RPN*.                                           COMCIDP     5961     A
COMCIDP  *                                                                                 COMCIDP     5962     A
COMCIDP  *                             PUSH TOKEN ON STACK.                                COMCIDP     5963     A
COMCIDP  *                                                                                 COMCIDP     5964     A
COMCIDP  *                   LEFT (:   PUSH ON STACK.                                      COMCIDP     5965     A
COMCIDP  *                                                                                 COMCIDP     5966     A
COMCIDP  *                   RIGHT ):  POP STACK THRU FIRST ( HIT.                         COMCIDP     5967     A
COMCIDP  *                                                                                 COMCIDP     5968     A
COMCIDP  *                   EOS:      POP ENTIRE STACK.                                   COMCIDP     5969     A
COMCIDP  *                                                                                 COMCIDP     5970     A
COMCIDP  *                                                                                 COMCIDP     5971     A
COMCIDP  *         POLISH IN --                                                            COMCIDP     5972     A
COMCIDP  *                                                                                 COMCIDP     5973     A
COMCIDP  *                WHILE NOT EOS                                                    COMCIDP     5974     A
COMCIDP  *                                                                                 COMCIDP     5975     A
COMCIDP  *                   IF TOKEN = OPERAND                                            COMCIDP     5976     A
COMCIDP  *                      PUSH ON STACK.                                             COMCIDP     5977     A
COMCIDP  *                                                                                 COMCIDP     5978     A
COMCIDP  *                   IF TOKEN = BINARY OPERATOR                                    COMCIDP     5979     A
COMCIDP  *                      APPLY OPERATOR ON TOP 2 STACK ENTRIES                      COMCIDP     5980     A
COMCIDP  *                      PUSH RESULT ON STACK (OVERLAYING TOP 2).                   COMCIDP     5981     A
COMCIDP  *                                                                                 COMCIDP     5982     A
COMCIDP  *                   IF TOKEN = UNARY OPERATOR                                     COMCIDP     5983     A
COMCIDP  *                      APPLY ON TOP STACK ENTRY                                   COMCIDP     5984     A
COMCIDP  *                      PUSH RESULT ON STACK (OVERLAYING TOP 1).                   COMCIDP     5985     A
COMCIDP  *                                                                                 COMCIDP     5986     A
COMCIDP  *                AT END, RESULT WILL BE IN STACK.                                 COMCIDP     5987     A
COMCIDP  *                                                                                 COMCIDP     5988     A
COMCIDP  *         THROUGHOUT THIS ROUTINE, REGISTERS B6 AND B7 HAVE GLOBAL                COMCIDP     5989     A
COMCIDP  *         SIGNIFICANCE --                                                         COMCIDP     5990     A
COMCIDP  *                                                                                 COMCIDP     5991     A
COMCIDP  *                (B6) = TOP-OF-STACK, *PAST*, ORDINAL.                            COMCIDP     5992     A
COMCIDP  *                (B7) = NEXT ADDR TO STORE/READ IN *RPN*.                         COMCIDP     5993     A
COMCIDP  *                                                                                 COMCIDP     5994     A
COMCIDP  *                                                                                 COMCIDP     5995     A
COMCIDP  *         ENTRY  (A5+1) = ADDR OF 1ST TOKEN TO PARSE.                             COMCIDP     5996     A
COMCIDP  *                                                                                 COMCIDP     5997     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *         EXIT   (IDPACC) = BINARY VALUE OF EXPRESSION.                COMCIDP    5998    A
COMCIDP  *                        = +0 IF NULL PARSE, (X3) .ZR.                COMCIDP    5999    A
COMCIDP  *                (X6)    = (IDPACC).                                  COMCIDP    6000    A
COMCIDP  *                (X1)    = (IDPACC).                                  COMCIDP    6001    A
COMCIDP  *                (X3)    = .NZ. IF EXPRESSION WAS NON-NULL, ELSE      COMCIDP    6002    A
COMCIDP  *                        = .ZR. IF EXPRESSION WAS NULL.               COMCIDP    6003    A
COMCIDP  *                (A5,X5) = A+C OF TOKEN THAT TERMINATED PARSE.        COMCIDP    6004    A
COMCIDP  *                                                                     COMCIDP    6005    A
COMCIDP  *                (PAS=TET) = (MX=TET).  SEE *MX=TET* FOR DETAILS.     COMCIDP    6006    A
COMCIDP  *                                                                     COMCIDP    6007    A
COMCIDP  *         USES   ALL                                                  COMCIDP    6008    A
COMCIDP  *                                                                     COMCIDP    6009    A
COMCIDP  *         CALLS  CHK,CIB,FAA,FRA,PTR,SSY                              COMCIDP    6010    A
COMCIDP                                                                        COMCIDP    6011    A
COMCIDP                                                                        COMCIDP    6012    A
COMCIDP   PAS      SUBR              ** ENTRY/EXIT **                          COMCIDP    6013    A
COMCIDP                                                                        COMCIDP    6014    A
COMCIDP  *         INITIALIZATION.                                             COMCIDP    6015    A
COMCIDP                                                                        COMCIDP    6016    A
COMCIDP            SX6    0                                                    COMCIDP    6017    A
COMCIDP            SA6    IDPACC      CLEAR ACCUMULATOR                        COMCIDP    6018    A
COMCIDP            SA6    IDPNUL      INITIALIZE TO *SUBEXPRESSION IS NULL*    COMCIDP    6019    A
COMCIDP                                                                        COMCIDP    6020    A
COMCIDP            SA1    TB=STMT     (X1) = ADDR OF STMT INFO WORD            COMCIDP    6021    A
COMCIDP            SA2    X1          (X2) = STMT INFO WORD FOR STMT TO PARSE  COMCIDP    6022    A
COMCIDP            ERRNZ  TB.LENL-18                                           COMCIDP    6023    A
COMCIDP            LX2    0-TB.LENP                                            COMCIDP    6024    A
COMCIDP            SB2    X2          (B2) = NR OF WORDS/TOKENS IN THIS STMT   COMCIDP    6025    A
COMCIDP            SB3    L.PAST      (B3) = NR OF AVAILABLE WORDS IN *PAST*   COMCIDP    6026    A
COMCIDP            GT     B2,B3,E.STB IF STMT TOO BIG, ERROR...                COMCIDP    6027    A
COMCIDP                                                                        COMCIDP    6028    A
COMCIDP            SB6    -1          INITIALIZE (B6) = *PAST* ORDINAL         COMCIDP    6029    A
COMCIDP            SB7    FW.RPN      (B7) = FWA OF *RPN* TABLE                COMCIDP    6030    A
COMCIDP   PAS=PO   SPACE  4,10                                                 COMCIDP    6031    A
COMCIDP  **        PAS=PO - POLISH OUT MAIN LOOP NODE.                         COMCIDP    6032    A
COMCIDP  *                                                                     COMCIDP    6033    A
COMCIDP  *         THIS IS THE MAIN LOOP NODE FOR GENERATION OF THE *RPN*      COMCIDP    6034    A
COMCIDP  *         (REVERSE POLISH NOTATION) TABLE, AND IS TRAVERSED ONCE FOR  COMCIDP    6035    A
COMCIDP  *         EACH TOKEN/OPERATOR/OPERAND ENCOUNTERED IN *TB* (TOKEN      COMCIDP    6036    A
COMCIDP  *         BUFFER).                                                    COMCIDP    6037    A
COMCIDP  *                                                                     COMCIDP    6038    A
COMCIDP  *         *PAS=PO* CAN BE THOUGHT OF AS HAVING 2 DISTINCT FUNCTIONS --COMCIDP    6039    A
COMCIDP  *                                                                     COMCIDP    6040    A
COMCIDP  *           1. EVALUATION OF OPERANDS INTO BINARY.                    COMCIDP    6041    A
COMCIDP  *                                                                     COMCIDP    6042    A
COMCIDP  *           2. ORDERING OF OPERATORS.                                 COMCIDP    6043    A
COMCIDP  *                                                                     COMCIDP    6044    A
COMCIDP  *         ENTRY  (B6) = ORDINAL IN *PAST* OF TOP-OF-STACK.            COMCIDP    6045    A
COMCIDP  *                (B7) = NEXT ADDR TO STORE INTO *RPN*.                COMCIDP    6046    A
COMCIDP  *                                                                     COMCIDP    6047    A
COMCIDP  *         NOTE THAT (B6) AND (B7) ARE SAVED AT THE TOP OF THIS LOOP IN COMCIDP    6048    A
COMCIDP  *         (PAS=SB6) AND (PAS=SB7), RESPECTIVELY.  THIS ALLOWS ANY POLISH COMCIDP  6049    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *         OUT PROCESSOR TO DESTROY (B6) AND (B7) WITHOUT HAVING TO SAVE          COMCIDP     6050     A
COMCIDP  *         THEM.  SAID PROCESSORS, HOWEVER, **MUST** RESTORE THESE                COMCIDP     6051     A
COMCIDP  *         REGISTERS BEFORE RETURNING TO *PAS=PO*.                                COMCIDP     6052     A
COMCIDP                                                                                   COMCIDP     6053     A
COMCIDP                                                                                   COMCIDP     6054     A
COMCIDP   PAS=PO   BSS    0           ** POLISH OUT MAIN LOOP NODE **                     COMCIDP     6055     A
COMCIDP            SA1    PAS=TET     (X1) = TERMINATING *TOT* SHIFT MASK                 COMCIDP     6056     A
COMCIDP            SA2    IDPNUL                                                          COMCIDP     6057     A
COMCIDP            SA5    A5+B1       (A5,X5) = A+C OF NEXT TOKEN                         COMCIDP     6058     A
COMCIDP            SB2    X5                                                              COMCIDP     6059     A
COMCIDP            LX1    B2                                                              COMCIDP     6060     A
COMCIDP            BX7    X5                                                              COMCIDP     6061     A
COMCIDP            SX6    X2+B1                                                           COMCIDP     6062     A
COMCIDP            MI     X1,PAS30    IF TERMINATING TOT, POP ENTIRE STACK...             COMCIDP     6063     A
COMCIDP            SA6    A2                                                              COMCIDP     6064     A
COMCIDP            SX6    B6                                                              COMCIDP     6065     A
COMCIDP            SX7    B7                                                              COMCIDP     6066     A
COMCIDP            SA6    PAS=SB6     SAVE (B6)                                           COMCIDP     6067     A
COMCIDP            SA7    PAS=SB7     SAVE (B7)                                           COMCIDP     6068     A
COMCIDP                                                                                   COMCIDP     6069     A
COMCIDP  *         GET *POT* ENTRY.                                                       COMCIDP     6070     A
COMCIDP                                                                                   COMCIDP     6071     A
COMCIDP            SB2    X5+                                                             COMCIDP     6072     A
COMCIDP            SB3    L.POT                                                           COMCIDP     6073     A
COMCIDP            LE     B2,B0,E.POP IF UNDER-SHOOTING *POT*, WOWY...                    COMCIDP     6074     A
COMCIDP            GT     B2,B3,E.POP IF OVER-SHOOTING *POT*, KAZOWY...                   COMCIDP     6075     A
COMCIDP                                                                                   COMCIDP     6076     A
COMCIDP            SA1    FW.POT+X5-O.VAR  (X1) = *POT* ENTRY FOR THIS OPERATOR           COMCIDP     6077     A
COMCIDP            LX1    0-PS.POPP                                                       COMCIDP     6078     A
COMCIDP            ERRNZ  PS.POPL-18                                                      COMCIDP     6079     A
COMCIDP            SB2    X1          (B2) = POLISH OUT PROCESSOR ADDR                    COMCIDP     6080     A
COMCIDP                                                                                   COMCIDP     6081     A
COMCIDP  *         EXTRACT *POT* INFORMATION.                                             COMCIDP     6082     A
COMCIDP                                                                                   COMCIDP     6083     A
COMCIDP            ERRNZ  TB.TOTL-18                                                      COMCIDP     6084     A
COMCIDP            ERRNZ  PS.TOTL-18                                                      COMCIDP     6085     A
COMCIDP            ERRNZ  PS.POPL-TB.TOTL                                                 COMCIDP     6086     A
COMCIDP            ERRNZ  PS.POPL-18                                                      COMCIDP     6087     A
COMCIDP            ERRNZ  PS.UPIPL-18                                                     COMCIDP     6088     A
COMCIDP            ERRNZ  PS.PIPL-18                                                      COMCIDP     6089     A
COMCIDP                                                                                   COMCIDP     6090     A
COMCIDP            LX1    0-PS.PIPP+PS.POPP-0                                             COMCIDP     6091     A
COMCIDP            SX2    X1          (X2) = *PIP*                                        COMCIDP     6092     A
COMCIDP            LX1    0-PS.UPIPP+PS.PIPP-0                                            COMCIDP     6093     A
COMCIDP            SX3    X1          (X3) = *UPIP*                                       COMCIDP     6094     A
COMCIDP            LX1    0-PS.PRIP+PS.UPIPP-0                                            COMCIDP     6095     A
COMCIDP            MX6    -PS.PRIL                                                        COMCIDP     6096     A
COMCIDP            SX7    X5          (X7) = *TOT*                                        COMCIDP     6097     A
COMCIDP            BX6    -X6*X1      (X6) = *PRI*                                        COMCIDP     6098     A
COMCIDP                                                                                   COMCIDP     6099     A
COMCIDP  *         CHECK FOR UNARY OPERATOR.                                              COMCIDP     6100     A
COMCIDP                                                                                   COMCIDP     6101     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            SA4    A5-B1       (X4) = PRECEDING TOKEN                          COMCIDP    6102    A
COMCIDP            SA1    MX=PREUN    (X1) = PRE-UNARY *TOT* SHIFT MASK               COMCIDP    6103    A
COMCIDP            SB3    X4                                                          COMCIDP    6104    A
COMCIDP            LX1    B3                                                          COMCIDP    6105    A
COMCIDP            PL     X1,PAS10    IF NOT A UNARY OPERATOR                         COMCIDP    6106    A
COMCIDP                                                                               COMCIDP    6107    A
COMCIDP            SX4    X5-O.SEP                                                    COMCIDP    6108    A
COMCIDP            MI     X4,PAS10    IF NOT AN OPERATOR                              COMCIDP    6109    A
COMCIDP                                                                               COMCIDP    6110    A
COMCIDP            SX4    X5-O.LP                                                     COMCIDP    6111    A
COMCIDP            ZR     X4,PAS10    IF (, LEAVE THINGS ALONE...                     COMCIDP    6112    A
COMCIDP                                                                               COMCIDP    6113    A
COMCIDP            SX6    77B         (X6) = UNARY OPERATOR PRIORITY                  COMCIDP    6114    A
COMCIDP            SB2    PAS22       (B2) = ADDR TO FORCE UNARY ON STACK             COMCIDP    6115    A
COMCIDP            SX2    X3+         *PIP* = *POP*                                   COMCIDP    6116    A
COMCIDP                                                                               COMCIDP    6117    A
COMCIDP  *         CONSTRUCT *PAST* ENTRY.                                            COMCIDP    6118    A
COMCIDP  *                                                                            COMCIDP    6119    A
COMCIDP  *         ENTRY  (X2) = *PIP*, POLISH IN PROCESSOR ADDR                      COMCIDP    6120    A
COMCIDP  *                (X6) = *PRI*, PARSING PRIORITY                              COMCIDP    6121    A
COMCIDP  *                (X7) = *TOT*, TOKEN TYPE                                    COMCIDP    6122    A
COMCIDP  *                (B2) = POLISH OUT PROCESSOR ADDR                            COMCIDP    6123    A
COMCIDP                                                                               COMCIDP    6124    A
COMCIDP   PAS10    LX7    PS.TOTP-0                                                   COMCIDP    6125    A
COMCIDP            LX2    PS.PIPP-0                                                   COMCIDP    6126    A
COMCIDP            BX4    X6          (X4) = *PRI*, PARSING PRIORITY                  COMCIDP    6127    A
COMCIDP            LX6    PS.PRIP-0                                                   COMCIDP    6128    A
COMCIDP            BX7    X7+X2       MERGE *PIP*                                     COMCIDP    6129    A
COMCIDP            BX7    X7+X6       MERGE *PRI*                                     COMCIDP    6130    A
COMCIDP                                                                               COMCIDP    6131    A
COMCIDP            JP     B2                                                          COMCIDP    6132    A
COMCIDP   PAS=PO   SPACE  4,10                                                        COMCIDP    6133    A
COMCIDP  **        POLISH OUT PROCESSORS.                                             COMCIDP    6134    A
COMCIDP  *                                                                            COMCIDP    6135    A
COMCIDP  *         FOLLOWING ARE THE POLISH OUT PROCESSORS THAT ARE RESPONSIBLE       COMCIDP    6136    A
COMCIDP  *         FOR GENERATION OF THE *RPN* (REVERSE POLISH NOTATION) TABLE.       COMCIDP    6137    A
COMCIDP  *                                                                            COMCIDP    6138    A
COMCIDP  *         ENTRY  (B6) = *PAST* TOP-OF-STACK ORDINAL.                         COMCIDP    6139    A
COMCIDP  *                (B7) = NEXT *RPN* ADDR TO STORE AT.                         COMCIDP    6140    A
COMCIDP  *                (X4) = PARSING PRIORITY OF THIS OPERATOR/OPERAND            COMCIDP    6141    A
COMCIDP  *                       (IGNORED FOR OPERANDS).                              COMCIDP    6142    A
COMCIDP  *                (X7) = CONSTRUCTED *PAST/RPN* ENTRY FOR THIS OPERATOR/      COMCIDP    6143    A
COMCIDP  *                       OPERAND.                                             COMCIDP    6144    A
COMCIDP  *                (A5,X5) = A+C OF CURRENT TOKEN.                            COMCIDP    6145    A
COMCIDP   PAS=PO   SPACE  4,10                                                        COMCIDP    6146    A
COMCIDP  *         HERE IF ( ENCOUNTERED -- PLACE ON STACK.                           COMCIDP    6147    A
COMCIDP                                                                               COMCIDP    6148    A
COMCIDP   PAS16    SB6    B6+1                                                        COMCIDP    6149    A
COMCIDP            SA7    FW.PAST+B6                                                  COMCIDP    6150    A
COMCIDP            EQ     PAS=PO      CONTINUE...                                     COMCIDP    6151    A
COMCIDP                                                                               COMCIDP    6152    A
COMCIDP                                                                               COMCIDP    6153    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *         HERE IF ) ENCOUNTERED -- POP *PAST* UNTIL ( ENCOUNTERED.
COMCIDP
COMCIDP   PAS18    SA1    FW.PAST+B6  (X1) = TOP OF STACK
COMCIDP            LT     B6,B0,PAS=PO IF *PAST* IS EMPTY, DONE...
COMCIDP            SX2    X1-O.LP
COMCIDP            BX6    X1
COMCIDP            SB6    B6-B1
COMCIDP            ZR     X2,PAS=PO   IF ( ENCOUNTERED, DONE...
COMCIDP            SA6    B7
COMCIDP            SB7    B7+B1
COMCIDP            EQ     PAS18
COMCIDP
COMCIDP
COMCIDP  *         HERE FOR ALL OPERATORS (NOT LEFT AND RIGHT PARENS).
COMCIDP  *
COMCIDP  *           1. WHILE PRIORITY OF *TB* OPERATOR .LE. PRIORITY OF TOP OF
COMCIDP  *              *PAST* OPERATOR, PASS TOP OF *PAST* OPERATOR TO *RPN*.
COMCIDP  *
COMCIDP  *           2. PLACE *TB* OPERATOR ON TOP OF *PAST*.
COMCIDP
COMCIDP   PAS20    SA1    FW.PAST+B6  (X1) = TOP OF STACK
COMCIDP            LT     B6,B0,PAS22 IF *PAST* IS EMPTY, DONE...
COMCIDP            MX2    -PS.PRIL
COMCIDP            BX6    X1
COMCIDP            LX1    0-PS.PRIP
COMCIDP            BX3    -X2*X1      (X3) = *PAST* TOP-OF-STACK OP PRIORITY
COMCIDP            IX1    X3-X4
COMCIDP            MI     X1,PAS22    IF *TB* OP PRIOR .GT. *PAST* OP PRIOR
COMCIDP            SA6    B7
COMCIDP            SB7    B7+B1
COMCIDP            SB6    B6-B1
COMCIDP            EQ     PAS20
COMCIDP
COMCIDP  *         PLACE *TB* OPERATOR ON STACK.
COMCIDP
COMCIDP   PAS22    SB6    B6+1
COMCIDP            SA7    FW.PAST+B6
COMCIDP            EQ     PAS=PO      CONTINUE...
COMCIDP
COMCIDP
COMCIDP  *         HERE IF *VAR* TOKEN.
COMCIDP  *
COMCIDP  *           1. CALL *SSY* (SEARCH SYMBOL TABLES) TO ASSOCIATE A BINARY
COMCIDP  *              VALUE WITH THIS *VAR* TOKEN.
COMCIDP  *
COMCIDP  *           2. PUT *PS* ENTRY FOR THIS *VAR* AND ITS EVALUATED BINARY
COMCIDP  *              VALUE OUT TO *RPN* AS A 2 WORD ENTRY.
COMCIDP
COMCIDP   PAS24    SA1    A5+1
COMCIDP            SX6    X1-O.VAR
COMCIDP            ZR     X6,E.SYTL   IF SYMBOL TOO LONG, ERROR...
COMCIDP
```

```
         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            SA7    B7                                                    COMCIDP    6206    A
COMCIDP            MX2    TB.TOCL                                               COMCIDP    6207    A
COMCIDP            BX1    X2*X5        (X1) = SYMBOL TO SEARCH FOR, -L- FMT     COMCIDP    6208    A
COMCIDP            RJ     SSY          SEARCH SYMBOL TABLES                     COMCIDP    6209    A
COMCIDP            ZR     B6,E.VAR     IF NO FIND                               COMCIDP    6210    A
COMCIDP                                                                         COMCIDP    6211    A
COMCIDP            SA1    PAS=SB6                                               COMCIDP    6212    A
COMCIDP            SA2    A1+B1                                                 COMCIDP    6213    A
COMCIDP            SB6    X1           RESTORE (B6)                             COMCIDP    6214    A
COMCIDP            SB7    X2+B1        (B7) = NEXT *RPN* ADDR TO STORE AT       COMCIDP    6215    A
COMCIDP            SA6    B7                                                    COMCIDP    6216    A
COMCIDP            SB7    B7+B1                                                 COMCIDP    6217    A
COMCIDP            EQ     PAS=PO       CONTINUE WITH POLISH OUT...              COMCIDP    6218    A
COMCIDP                                                                         COMCIDP    6219    A
COMCIDP                                                                         COMCIDP    6220    A
COMCIDP  *         HERE IF *CONS* TOKEN.                                        COMCIDP    6221    A
COMCIDP  *                                                                      COMCIDP    6222    A
COMCIDP  *           1. CALL *CIB* (CONVERT INTEGER TO BINARY) TO EVALUATE *CONS*  COMCIDP  6223  A
COMCIDP  *              TOKEN(S).                                               COMCIDP    6224    A
COMCIDP  *                                                                      COMCIDP    6225    A
COMCIDP  *           2. PUT *PS* ENTRY FOR THIS *CONS* AND ITS EVALUATED BINARY COMCIDP    6226    A
COMCIDP  *              OUT TO *RPN* AS A 2 WORD ENTRY.                         COMCIDP    6227    A
COMCIDP                                                                         COMCIDP    6228    A
COMCIDP   PAS26    SA7    B7                                                    COMCIDP    6229    A
COMCIDP            SB7    B7+B1                                                 COMCIDP    6230    A
COMCIDP            SX1    8            SET TO *DEFAULT BASE IS OCTAL*           COMCIDP    6231    A
COMCIDP            RJ     CIB          CONVERT INTEGER TO BINARY                COMCIDP    6232    A
COMCIDP            NZ     B2,E.CON     IF AN ERROR DURING CONVERSION            COMCIDP    6233    A
COMCIDP                                                                         COMCIDP    6234    A
COMCIDP            SA6    B7                                                    COMCIDP    6235    A
COMCIDP            SB7    B7+B1                                                 COMCIDP    6236    A
COMCIDP            EQ     PAS=PO                                                COMCIDP    6237    A
COMCIDP                                                                         COMCIDP    6238    A
COMCIDP                                                                         COMCIDP    6239    A
COMCIDP  *         HERE IF *REG* TOKEN.                                         COMCIDP    6240    A
COMCIDP  *                                                                      COMCIDP    6241    A
COMCIDP  *           1. GET ADDR OF SAVED REGISTER FROM IDP'S SAVED USER REGISTER  COMCIDP  6242  A
COMCIDP  *              AREA.                                                   COMCIDP    6243    A
COMCIDP  *                                                                      COMCIDP    6244    A
COMCIDP  *           2. PUT *PS* ENTRY FOR THIS *REG* AND APPROPRIATE REGISTER  COMCIDP    6245    A
COMCIDP  *              ADDR OUT TO *RPN* AS A 2 WORD ENTRY.                    COMCIDP    6246    A
COMCIDP                                                                         COMCIDP    6247    A
COMCIDP   PAS28    LX5    0-TB.0TRP                                             COMCIDP    6248    A
COMCIDP            ERRNZ  TB.0TRL-18                                            COMCIDP    6249    A
COMCIDP            SX6    IDP=SVB+X5   (X6) = ADDR OF SAVED REGISTER            COMCIDP    6250    A
COMCIDP            SA7    B7                                                    COMCIDP    6251    A
COMCIDP            SA6    A7+B1                                                 COMCIDP    6252    A
COMCIDP            SB7    B7+2                                                  COMCIDP    6253    A
COMCIDP            EQ     PAS=PO                                                COMCIDP    6254    A
COMCIDP                                                                         COMCIDP    6255    A
COMCIDP                                                                         COMCIDP    6256    A
COMCIDP  *         HERE IF AN EOS ENCOUNTERED.                                  COMCIDP    6257    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *                                                                  COMCIDP    6258    A
COMCIDP  *         POP ENTIRE STACK.                                        COMCIDP    6259    A
COMCIDP                                                                     COMCIDP    6260    A
COMCIDP   PAS30    SA1    FW.PAST+B6  (X1) = TOP OF STACK                   COMCIDP    6261    A
COMCIDP            LT     B6,B0,PAS32 IF DONE                               COMCIDP    6262    A
COMCIDP            SX2    X1-O.LP                                           COMCIDP    6263    A
COMCIDP            BX6    X1                                                COMCIDP    6264    A
COMCIDP            SB6    B6-B1                                             COMCIDP    6265    A
COMCIDP            ZR     X2,PAS30    IF (, IGNORE...                       COMCIDP    6266    A
COMCIDP            SA6    B7                                                COMCIDP    6267    A
COMCIDP            SB7    B7+B1                                             COMCIDP    6268    A
COMCIDP            EQ     PAS30                                             COMCIDP    6269    A
COMCIDP                                                                     COMCIDP    6270    A
COMCIDP                                                                     COMCIDP    6271    A
COMCIDP  *         HERE WHEN WE HAVE COMPLETED GENERATING *RPN* TABLE.      COMCIDP    6272    A
COMCIDP  *                                                                  COMCIDP    6273    A
COMCIDP  *           1. TERMINATE *RPN* TABLE WITH A FULL ZERO END-OF-TABLE COMCIDP    6274    A
COMCIDP  *              MARK.                                               COMCIDP    6275    A
COMCIDP  *                                                                  COMCIDP    6276    A
COMCIDP  *           2. CHECK FOR *RPN* TABLE OVERFLOW.                     COMCIDP    6277    A
COMCIDP  *                                                                  COMCIDP    6278    A
COMCIDP  *           3. SET UP FOR *POLISH IN* PROCESSING.                  COMCIDP    6279    A
COMCIDP                                                                     COMCIDP    6280    A
COMCIDP   PAS32    BX6    X6-X6                                             COMCIDP    6281    A
COMCIDP            SA6    B7          MARK EOT                              COMCIDP    6282    A
COMCIDP                                                                     COMCIDP    6283    A
COMCIDP            SB2    FW.RPN+L.RPN  (B2) = LWA+1 OF AVAILABLE *RPN* SPACE COMCIDP    6284    A
COMCIDP            GE     B7,B2,SE.RPN  IF *RPN* OVERFLOW                   COMCIDP    6285    A
COMCIDP                                                                     COMCIDP    6286    A
COMCIDP            SB6    -1          INITIALIZE (B6) = *PAST* ORDINAL      COMCIDP    6287    A
COMCIDP            SB7    FW.RPN      (B7) = FWA OF *RPN* TABLE             COMCIDP    6288    A
COMCIDP   PAS=PI   SPACE  4,10                                              COMCIDP    6289    A
COMCIDP  **        PAS=PI - POLISH IN MAIN LOOP NODE.                       COMCIDP    6290    A
COMCIDP  *                                                                  COMCIDP    6291    A
COMCIDP  *         THIS IS THE MAIN LOOP NODE FOR THE EVALUATION OF THE *RPN* COMCIDP    6292    A
COMCIDP  *         (REVERSE POLISH NOTATION) TABLE INTO A SINGLE BINARY VALUE. COMCIDP    6293    A
COMCIDP  *                                                                  COMCIDP    6294    A
COMCIDP  *         *PAS=PI* IS TRAVERSED ONCE FOR EACH OPERATOR/OPERAND ENTRY COMCIDP    6295    A
COMCIDP  *         ENCOUNTERED IN *RPN*.                                    COMCIDP    6296    A
COMCIDP  *                                                                  COMCIDP    6297    A
COMCIDP  *         ENTRY  (B6) = *PAST* TOP-OF-STACK ORDINAL.               COMCIDP    6298    A
COMCIDP  *                (B7) = NEXT *RPN* ADDR TO LOOK AT.                COMCIDP    6299    A
COMCIDP  *                (A5,X5) = A+C OF TOKEN THAT TERMINATED PARSE (UNUSED COMCIDP    6300    A
COMCIDP  *                        DURING POLISH IN PROCESSING).             COMCIDP    6301    A
COMCIDP  *                                                                  COMCIDP    6302    A
COMCIDP  *         NOTE THAT (B6) AND (B7) ARE SAVED AT THE TOP OF THIS LOOP IN COMCIDP    6303    A
COMCIDP  *         (PAS=SB6) AND (PAS=SB7), RESPECTIVELY.  THIS ALLOWS ANY POLISH COMCIDP    6304    A
COMCIDP  *         IN PROCESSOR TO DESTROY (B6) AND (B7) WITHOUT HAVING TO SAVE COMCIDP    6305    A
COMCIDP  *         THEM.  SAID PROCESSORS, HOWEVER, **MUST** RESTORE THESE  COMCIDP    6306    A
COMCIDP  *         REGISTERS BEFORE RETURNING TO *PAS=PI*.                  COMCIDP    6307    A
COMCIDP                                                                     COMCIDP    6308    A
COMCIDP                                                                     COMCIDP    6309    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP   PAS=PI   BSS    0           ** POLISH IN MAIN LOOP NODE **
COMCIDP            SA1    B7          (X1) = NEXT *RPN* ENTRY
COMCIDP            SB7    B7+B1
COMCIDP            ZR     X1,PAS=XIT  IF *RPN* DEPLETED, DONE...
COMCIDP
COMCIDP            SX6    B6
COMCIDP            SX7    B7
COMCIDP            SA6    PAS=SB6     SAVE (B6)
COMCIDP            SA7    PAS=SB7     SAVE (B7)
COMCIDP
COMCIDP            LX1    0-PS.PIPP
COMCIDP            ERRNZ  PS.PIPL-18
COMCIDP            SB2    X1          (B2) = POLISH IN PROCESSOR ADDR
COMCIDP
COMCIDP            SA1    FW.PAST+B6    (X1) = A
COMCIDP            SA2    FW.PAST+B6-1  (X2) = B
COMCIDP            JP     B2          PARSE ON, PARSE ON...
COMCIDP   PAS=PI   SPACE  4,10
COMCIDP   **       POLISH IN PROCESSORS.
COMCIDP   *
COMCIDP   *        FOLLOWING ARE THE POLISH IN PROCESSORS THAT ARE RESPONSIBLE
COMCIDP   *        FOR ACTUALLY PERFORMING THE OPERATIONS SPECIFIED VIA IDP'S
COMCIDP   *        OPERATORS UPON THE OPERANDS.
COMCIDP   *
COMCIDP   *        ENTRY  (B6) = *PAST* TOP-OF-STACK ORDINAL.
COMCIDP   *               (B7) = NEXT *RPN* ADDR TO LOOK AT.
COMCIDP   *               (X1) = OPERAND -A-, TOP-OF-STACK.
COMCIDP   *               (X2) = OPERAND -B-, NEXT-TO-TOP-OF-STACK.
COMCIDP   *
COMCIDP   *                      THAT IS, OPERATIONS ARE --
COMCIDP   *
COMCIDP   *                        1. UNARY: OPERATE -A-.
COMCIDP   *
COMCIDP   *                        2. BINARY: -B- OPERATE -A-.
COMCIDP   PAS=PI   SPACE  4,10
COMCIDP   *        HERE IF --
COMCIDP   *
COMCIDP   *          1. *VAR* TOKEN.
COMCIDP   *
COMCIDP   *          2. *CONS* TOKEN.
COMCIDP   *
COMCIDP   *          3. *REG* TOKEN.
COMCIDP   *
COMCIDP   *        PUSH BINARY ON STACK.
COMCIDP
COMCIDP   PAS34    SA1    B7
COMCIDP            SB7    B7+B1
COMCIDP            BX6    X1
COMCIDP            SB6    B6+B1
COMCIDP            SA6    FW.PAST+B6
COMCIDP            EQ     PAS=PI
COMCIDP
```

```
          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP                                                                                COMCIDP     6362     A
COMCIDP  *         HERE IF: B+A                                                        COMCIDP     6363     A
COMCIDP                                                                                COMCIDP     6364     A
COMCIDP   PAS36    IX7    X2+X1       (X7) = B+A                                       COMCIDP     6365     A
COMCIDP            EQ     PAS=PIB                                                     COMCIDP     6366     A
COMCIDP                                                                                COMCIDP     6367     A
COMCIDP                                                                                COMCIDP     6368     A
COMCIDP  *         HERE IF: +A                                                         COMCIDP     6369     A
COMCIDP  *                                                                             COMCIDP     6370     A
COMCIDP  *         DETERMINE CURRENT DECK FWA BY --                                    COMCIDP     6371     A
COMCIDP  *                                                                             COMCIDP     6372     A
COMCIDP  *           1. GETTING CURRENT *CODE* 1ST DEFAULT POINTER.  THIS              COMCIDP     6373     A
COMCIDP  *              TELLS US WHERE OUR FRIEND IS *CODE*ING AT.                     COMCIDP     6374     A
COMCIDP  *                                                                             COMCIDP     6375     A
COMCIDP  *           2. CALLING *FRA* (FIND RELATIVE ADDRESS) WITH OUR                 COMCIDP     6376     A
COMCIDP  *              *CODE* POINTER IN ORDER TO GET THE DECK NAME                   COMCIDP     6377     A
COMCIDP  *              PLUS RELATIVE OFFSET.                                          COMCIDP     6378     A
COMCIDP  *                                                                             COMCIDP     6379     A
COMCIDP  *           3. CALLING *FAA* (FIND ABSOLUTE ADDRESS) WITH OUR                 COMCIDP     6380     A
COMCIDP  *              JUST-FOUND-BY-FRA DECK NAME IN ORDER TO FIND                   COMCIDP     6381     A
COMCIDP  *              OUT WHAT THE ABSOLUTE ADDRESS OF THIS DECK IS.                 COMCIDP     6382     A
COMCIDP  *                                                                             COMCIDP     6383     A
COMCIDP  *         THIS IS OUR DESIRED NUMBER.                                         COMCIDP     6384     A
COMCIDP                                                                                COMCIDP     6385     A
COMCIDP   PAS38    SB2    B1          SET TO *1ST DEFAULT*                             COMCIDP     6386     A
COMCIDP            SB3    -B1         SET TO *GET POINTER*                             COMCIDP     6387     A
COMCIDP            SA1    ST.COD      (X1) = *KEYW* ENTRY FOR *CODE*                   COMCIDP     6388     A
COMCIDP            RJ     PTR         GET POINTER                                      COMCIDP     6389     A
COMCIDP            SX1    X2          (X1) = DEFAULT *CODE* FWA                        COMCIDP     6390     A
COMCIDP            RJ     FRA         FIND RELATIVE ADDRESS                            COMCIDP     6391     A
COMCIDP            MI     B7,PAS39    IF USER DID NOT PROVIDE *FRA=*                   COMCIDP     6392     A
COMCIDP            BX1    X7          (X1) = DECK NAME, -L- FMT                        COMCIDP     6393     A
COMCIDP            MX0    7*CHAR                                                       COMCIDP     6394     A
COMCIDP            RJ     FAA         FIND ABSOLUTE ADDRESS                            COMCIDP     6395     A
COMCIDP            MI     X6,SE.UNPL  IF NO FIND                                       COMCIDP     6396     A
COMCIDP                                                                                COMCIDP     6397     A
COMCIDP   PAS39    SA1    PAS=SB6     (X1) = SAVED (B6)                                COMCIDP     6398     A
COMCIDP            SA2    A1+B1       (X2) = SAVED (B7)                                COMCIDP     6399     A
COMCIDP            SB6    X1          RESTORE (B6)                                     COMCIDP     6400     A
COMCIDP            SB7    X2          RESTORE (B7)                                     COMCIDP     6401     A
COMCIDP                                                                                COMCIDP     6402     A
COMCIDP            SA1    FW.PAST+B6  (X1) = A, RESTORED                               COMCIDP     6403     A
COMCIDP            IX7    X6+X1       (X7) = DECK+A                                    COMCIDP     6404     A
COMCIDP            EQ     PAS=PIU                                                      COMCIDP     6405     A
COMCIDP                                                                                COMCIDP     6406     A
COMCIDP                                                                                COMCIDP     6407     A
COMCIDP  *         HERE IF: B-A                                                        COMCIDP     6408     A
COMCIDP                                                                                COMCIDP     6409     A
COMCIDP   PAS40    IX7    X2-X1       (X7) = B-A                                       COMCIDP     6410     A
COMCIDP            EQ     PAS=PIB                                                      COMCIDP     6411     A
COMCIDP                                                                                COMCIDP     6412     A
COMCIDP                                                                                COMCIDP     6413     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *         HERE IF: -A                                          COMCIDP    6414    A
COMCIDP                                                                 COMCIDP    6415    A
COMCIDP   PAS42    BX7    -X1         (X7) = -A                         COMCIDP    6416    A
COMCIDP            EQ     PAS=PIU                                       COMCIDP    6417    A
COMCIDP                                                                 COMCIDP    6418    A
COMCIDP                                                                 COMCIDP    6419    A
COMCIDP  *         HERE IF: B*A                                         COMCIDP    6420    A
COMCIDP                                                                 COMCIDP    6421    A
COMCIDP   PAS44    IX7    X2*X1       (X7) = B*A                        COMCIDP    6422    A
COMCIDP            EQ     PAS=PIB                                       COMCIDP    6423    A
COMCIDP                                                                 COMCIDP    6424    A
COMCIDP                                                                 COMCIDP    6425    A
COMCIDP  *         HERE IF: *A                                          COMCIDP    6426    A
COMCIDP                                                                 COMCIDP    6427    A
COMCIDP   PAS46    SB2    X1                                            COMCIDP    6428    A
COMCIDP            RJ     CHK         CHECK CM ADDRESS                  COMCIDP    6429    A
COMCIDP            MI     B2,E.IND    IF ADDR IS BAD                    COMCIDP    6430    A
COMCIDP            SA1    B2                                            COMCIDP    6431    A
COMCIDP            BX7    X1          (X7) = *A                         COMCIDP    6432    A
COMCIDP            EQ     PAS=PIU                                       COMCIDP    6433    A
COMCIDP                                                                 COMCIDP    6434    A
COMCIDP                                                                 COMCIDP    6435    A
COMCIDP  *         HERE IF: B$A                                         COMCIDP    6436    A
COMCIDP                                                                 COMCIDP    6437    A
COMCIDP   PAS48    SB2    X1                                            COMCIDP    6438    A
COMCIDP            LX7    X2,B2       (X7) = B$A                        COMCIDP    6439    A
COMCIDP            EQ     PAS=PIB                                       COMCIDP    6440    A
COMCIDP                                                                 COMCIDP    6441    A
COMCIDP                                                                 COMCIDP    6442    A
COMCIDP  *         HERE IF: B.OR.A                                      COMCIDP    6443    A
COMCIDP                                                                 COMCIDP    6444    A
COMCIDP   PAS50    BX7    X2+X1       (X7) = B.OR.A                     COMCIDP    6445    A
COMCIDP            EQ     PAS=PIB                                       COMCIDP    6446    A
COMCIDP                                                                 COMCIDP    6447    A
COMCIDP                                                                 COMCIDP    6448    A
COMCIDP  *         HERE IF: B.AND.A                                     COMCIDP    6449    A
COMCIDP                                                                 COMCIDP    6450    A
COMCIDP   PAS52    BX7    X2*X1       (X7) = B.AND.A                    COMCIDP    6451    A
COMCIDP            EQ     PAS=PIB                                       COMCIDP    6452    A
COMCIDP                                                                 COMCIDP    6453    A
COMCIDP                                                                 COMCIDP    6454    A
COMCIDP  *         HERE IF: B.XOR.A                                     COMCIDP    6455    A
COMCIDP                                                                 COMCIDP    6456    A
COMCIDP   PAS54    BX7    X2-X1       (X7) = B.XOR.A                    COMCIDP    6457    A
COMCIDP            EQ     PAS=PIB                                       COMCIDP    6458    A
COMCIDP                                                                 COMCIDP    6459    A
COMCIDP                                                                 COMCIDP    6460    A
COMCIDP  *         HERE TO PUSH RESULT OF BINARY OPERATION ON STACK.    COMCIDP    6461    A
COMCIDP                                                                 COMCIDP    6462    A
COMCIDP   PAS=PIB  BSS    0                                             COMCIDP    6463    A
COMCIDP            SB6    B6-1                                          COMCIDP    6464    A
COMCIDP            LT     B6,B0,E.PAST IF STACK IS EMPTY, ERROR...      COMCIDP    6465    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            SA7    FW.PAST+B6                                          COMCIDP    6466    A
COMCIDP            EQ     PAS=PI      CONTINUE WITH POLISH IN...              COMCIDP    6467    A
COMCIDP                                                                       COMCIDP    6468    A
COMCIDP                                                                       COMCIDP    6469    A
COMCIDP  *         HERE TO PUSH RESULT OF UNARY OPERATION ON STACK.           COMCIDP    6470    A
COMCIDP                                                                       COMCIDP    6471    A
COMCIDP   PAS=PIU  BSS    0                                                   COMCIDP    6472    A
COMCIDP            SA7    FW.PAST+B6                                          COMCIDP    6473    A
COMCIDP            EQ     PAS=PI      CONTINUE WITH POLISH IN...              COMCIDP    6474    A
COMCIDP   PAS=XIT  SPACE  4,10                                                COMCIDP    6475    A
COMCIDP  **        PAS=XIT - HERE WHEN READY TO EXIT.                         COMCIDP    6476    A
COMCIDP  *                                                                    COMCIDP    6477    A
COMCIDP  *           1. CHECK TO MAKE SURE THAT THE RESULT OF OUR EVALUATION  COMCIDP    6478    A
COMCIDP  *              IS THE ONLY THING LEFT ON THE STACK. IF NOT THEN WE   COMCIDP    6479    A
COMCIDP  *              HAVE AN ERROR.                                        COMCIDP    6480    A
COMCIDP  *                                                                    COMCIDP    6481    A
COMCIDP  *           2. SET UP EXIT CONDITIONS, AND EXIT BACK TO CALLER.      COMCIDP    6482    A
COMCIDP                                                                       COMCIDP    6483    A
COMCIDP                                                                       COMCIDP    6484    A
COMCIDP   PAS=XIT  BSS    0                                                   COMCIDP    6485    A
COMCIDP            SA3    IDPNUL                                              COMCIDP    6486    A
COMCIDP            SA2    MX=TET      (X2) = MASTER TERMINATING *TOT* SHIFT MASK COMCIDP 6487    A
COMCIDP            SX1    0                                                   COMCIDP    6488    A
COMCIDP            ZR     X3,PAS90    IF NULL PARSE                           COMCIDP    6489    A
COMCIDP                                                                       COMCIDP    6490    A
COMCIDP            SA1    FW.PAST+B6  (X1) = BINARY RESULT                    COMCIDP    6491    A
COMCIDP            NZ     B6,E.PAST   IF OPERATOR/OPERAND MISMATCH, ERROR...  COMCIDP    6492    A
COMCIDP                                                                       COMCIDP    6493    A
COMCIDP   PAS90    BX6    X1                                                  COMCIDP    6494    A
COMCIDP            LX7    X2                                                  COMCIDP    6495    A
COMCIDP            SA6    IDPACC      SAVE ACCUMULATOR                        COMCIDP    6496    A
COMCIDP            SA7    PAS=TET                                             COMCIDP    6497    A
COMCIDP            EQ     EXIT.                                               COMCIDP    6498    A
COMCIDP                                                                       COMCIDP    6499    A
COMCIDP                                                                       COMCIDP    6500    A
COMCIDP   PASA     BSSZ   3                                                   COMCIDP    6501    A
COMCIDP                                                                       COMCIDP    6502    A
COMCIDP            LOC    PASA                                                COMCIDP    6503    A
COMCIDP   PAS=SB6  =      *           SAVED (B6)                              COMCIDP    6504    A
COMCIDP   PAS=SB7  =      *+1         SAVED (B7)                              COMCIDP    6505    A
COMCIDP   PAS=SX7  =      *+2         SAVED (X7)                              COMCIDP    6506    A
COMCIDP            LOC    *O                                                  COMCIDP    6507    A
COMCIDP                                                                       COMCIDP    6508    A
COMCIDP   PAS=TET  CON    "'?IDCTET"  LOCAL TERMINATING *TOT* SHIFT MASK      COMCIDP    6509    A
COMCIDP   PAT      SPACE  4,10                                                COMCIDP    6510    A
COMCIDP  **        PAT - PARSE *FWA,LWA,LEN* TRIPLE.                          COMCIDP    6511    A
COMCIDP  *                                                                    COMCIDP    6512    A
COMCIDP  *                                                                    COMCIDP    6513    A
COMCIDP  *         THIS ROUTINE CONTROLS THE PARSING OF THE *FWA,LWA,LEN*     COMCIDP    6514    A
COMCIDP  *         SYNTAX, AND IS USED MERELY AS A SPACE OPTIMIZATION BECAUSE COMCIDP    6515    A
COMCIDP  *         THIS SYNTAX NEEDS TO BE PARSED FOR A NUMBER OF DIFFERENT   COMCIDP    6516    A
COMCIDP  *         STATEMENTS.                                                COMCIDP    6517    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *                                                                  COMCIDP    6518    A
COMCIDP  *         ANY OR ALL OF THE SUBEXPRESSIONS WHICH CONSTITUTE THE    COMCIDP    6519    A
COMCIDP  *         *FWA,LWA,LEN* TRIPLE MAY BE NULL.  IF A NULL SUBEXPRESSION COMCIDP  6520    A
COMCIDP  *         IS ENCOUNTERED, *PAT* WILL USE A DEFAULT VALUE THAT HAS  COMCIDP    6521    A
COMCIDP  *         BEEN SET UP BY THE CALLER.                               COMCIDP    6522    A
COMCIDP  *                                                                  COMCIDP    6523    A
COMCIDP  *         NOTE THAT *PAT* RESOLVES ANY POSSIBLE SYNTACTIC DIFFERENCES COMCIDP 6524    A
COMCIDP  *         BETWEEN THE USES OF THE *LWA* AND *LEN* PARAMETERS --    COMCIDP    6525    A
COMCIDP  *                                                                  COMCIDP    6526    A
COMCIDP  *           1. FOR *FWA,LWA* (I.E. WHEN *LEN* IS NULL),            COMCIDP    6527    A
COMCIDP  *              NEED TO SET: LEN = LWA-FWA+1                        COMCIDP    6528    A
COMCIDP  *                                                                  COMCIDP    6529    A
COMCIDP  *           2. FOR *FWA,,LEN* (I.E. WHEN *LWA* IS NULL),           COMCIDP    6530    A
COMCIDP  *              NEED TO SET: LWA = FWA+LEN-1                        COMCIDP    6531    A
COMCIDP  *                                                                  COMCIDP    6532    A
COMCIDP  *         THIS HOCUS POCUS IS NECESSARY TO INSURE THAT *LWA* AND *LEN* COMCIDP 6533    A
COMCIDP  *         DO NOT CONFLICT WITH EACH OTHER.                         COMCIDP    6534    A
COMCIDP  *                                                                  COMCIDP    6535    A
COMCIDP  *         CONSIDER --                                              COMCIDP    6536    A
COMCIDP  *                                                                  COMCIDP    6537    A
COMCIDP  *               SNAP 100,,2                                        COMCIDP    6538    A
COMCIDP  *                                                                  COMCIDP    6539    A
COMCIDP  *         FOLLOWED BY --                                           COMCIDP    6540    A
COMCIDP  *                                                                  COMCIDP    6541    A
COMCIDP  *               SNAP 100,110                                       COMCIDP    6542    A
COMCIDP  *                                                                  COMCIDP    6543    A
COMCIDP  *         IF THE ABOVE *LWA/LEN* FIDDLING WERE NOT PERFORMED, UPON COMCIDP    6544    A
COMCIDP  *         ENCOUNTERING THE 2ND STMT, IDP WOULD NOT KNOW WHICH VALUE COMCIDP   6545    A
COMCIDP  *         TO USE: LEN=2 (RESIDUAL), OR LWA=110 (EXPLICIT).         COMCIDP    6546    A
COMCIDP  *                                                                  COMCIDP    6547    A
COMCIDP  *         ENTRY  (A5+1)    = ADDR OF 1ST TOKEN IN TRIPLE, I.E. *FWA*. COMCIDP 6548    A
COMCIDP  *                (TB=KEY)  = ADDR OF *KEYW* ENTRY, WORD 1, FOR THIS COMCIDP   6549    A
COMCIDP  *                            STMT (I.E. CONTAINS *PTR* ORD).       COMCIDP    6550    A
COMCIDP  *                                                                  COMCIDP    6551    A
COMCIDP  *         EXIT   (A5,X5)   = A+C OF TOKEN THAT TERMINATED TRIPLE   COMCIDP    6552    A
COMCIDP  *                (X2)      = FWA, AS SPECIFIED EXPLICITLY OR BY DEFAULT COMCIDP 6553    A
COMCIDP  *                (X3)      = LWA, AS SPECIFIED EXPLICITLY OR BY DEFAULT COMCIDP 6554    A
COMCIDP  *                (X4)      = LEN, AS SPECIFIED EXPLICITLY OR BY DEFAULT COMCIDP 6555    A
COMCIDP  *                (AP=FWA)  = (PAT#FWA) = FWA                       COMCIDP    6556    A
COMCIDP  *                (AP=LWA)  = LWA, AS COMPUTED FROM *LWA* AND/OR *LEN* COMCIDP 6557    A
COMCIDP  *                (B2)      = .MI. IF FWA, LWA, OR LEN ARE BAD, ELSE .PL. COMCIDP 6558    A
COMCIDP  *                                                                  COMCIDP    6559    A
COMCIDP  *         USES   ALL                                               COMCIDP    6560    A
COMCIDP  *                                                                  COMCIDP    6561    A
COMCIDP  *         CALLS  FLL,PAS,PTR                                       COMCIDP    6562    A
COMCIDP                                                                     COMCIDP    6563    A
COMCIDP                                                                     COMCIDP    6564    A
COMCIDP   PAT      SUBR               ** ENTRY/EXIT **                      COMCIDP    6565    A
COMCIDP                                                                     COMCIDP    6566    A
COMCIDP  *         INITIALIZATION.                                          COMCIDP    6567    A
COMCIDP                                                                     COMCIDP    6568    A
COMCIDP            SB2    B1          SET TO *1ST DEFAULT, PLEASE*          COMCIDP    6569    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP          SB3    -1          SET TO *GET PTR*                          COMCIDP    6570    A
COMCIDP          BX1    X1-X1       SET TO *USE KEYW ENTRY AT (TB=KEY)*       COMCIDP    6571    A
COMCIDP          RJ     PTR         GET POINTER VALUES                        COMCIDP    6572    A
COMCIDP                                                                       COMCIDP    6573    A
COMCIDP  *       SET UP DEFAULT FWA,LWA,LEN.                                  COMCIDP    6574    A
COMCIDP                                                                       COMCIDP    6575    A
COMCIDP          MX6    0                                                     COMCIDP    6576    A
COMCIDP          SA6    PATFLG      CLEAR *PATFLG*                            COMCIDP    6577    A
COMCIDP          SA6    PAT#LWA     DEFAULT LWA = 0 (I.E. NO LWA)             COMCIDP    6578    A
COMCIDP          BX6    X2                                                    COMCIDP    6579    A
COMCIDP          LX7    X4                                                    COMCIDP    6580    A
COMCIDP          SA6    PAT#FWA     DEFAULT FWA = (FWA POINTER)               COMCIDP    6581    A
COMCIDP          SA7    PAT#LEN     DEFAULT LEN = (LEN POINTER)               COMCIDP    6582    A
COMCIDP                                                                       COMCIDP    6583    A
COMCIDP  *       PARSE/PROCESS FWA.                                           COMCIDP    6584    A
COMCIDP                                                                       COMCIDP    6585    A
COMCIDP          ZR     X5,PAT4     IF EOS ENCOUNTERED                        COMCIDP    6586    A
COMCIDP          RJ     PAS         PARSE SUBEXPRESSION -- FWA                COMCIDP    6587    A
COMCIDP          ZR     X3,PAT2     IF FWA IS NULL                            COMCIDP    6588    A
COMCIDP                                                                       COMCIDP    6589    A
COMCIDP  *       CHECK TO SEE IF FWA CHANGED.  IF SO, SET DEFAULT LEN = 1.    COMCIDP    6590    A
COMCIDP                                                                       COMCIDP    6591    A
COMCIDP          SA2    PAT#FWA     (X2) = DEFAULT FWA                        COMCIDP    6592    A
COMCIDP          SX6    X1          (X6) = NEW FWA                            COMCIDP    6593    A
COMCIDP          SX7    B1                                                    COMCIDP    6594    A
COMCIDP          IX3    X6-X2                                                 COMCIDP    6595    A
COMCIDP          SA6    A2          UPDATE NEW FWA                            COMCIDP    6596    A
COMCIDP          ZR     X3,PAT2     IF FWA DID NOT CHANGE                     COMCIDP    6597    A
COMCIDP          SA7    PAT#LEN     SET TO DEFAULT LEN = 1                    COMCIDP    6598    A
COMCIDP                                                                       COMCIDP    6599    A
COMCIDP  *       PARSE/PROCESS LWA.                                           COMCIDP    6600    A
COMCIDP                                                                       COMCIDP    6601    A
COMCIDP   PAT2   ZR     X5,PAT4     IF EOS ENCOUNTERED                        COMCIDP    6602    A
COMCIDP          RJ     PAS         PARSE SUBEXPRESSION -- LWA                COMCIDP    6603    A
COMCIDP          ZR     X3,PAT3     IF LWA IS NULL                            COMCIDP    6604    A
COMCIDP          SX6    X1                                                    COMCIDP    6605    A
COMCIDP          SX7    PAF.LWAM                                              COMCIDP    6606    A
COMCIDP          SA6    PAT#LWA                                               COMCIDP    6607    A
COMCIDP          SA7    PATFLG                                                COMCIDP    6608    A
COMCIDP                                                                       COMCIDP    6609    A
COMCIDP  *       PARSE/PROCESS LEN.                                           COMCIDP    6610    A
COMCIDP                                                                       COMCIDP    6611    A
COMCIDP   PAT3   ZR     X5,PAT4     IF EOS ENCOUNTERED                        COMCIDP    6612    A
COMCIDP          RJ     PAS         PARSE SUBEXPRESSION -- LEN                COMCIDP    6613    A
COMCIDP          ZR     X3,PAT4     IF LEN IS NULL                            COMCIDP    6614    A
COMCIDP          SA2    PATFLG                                                COMCIDP    6615    A
COMCIDP          SX7    PAF.LENM                                              COMCIDP    6616    A
COMCIDP          SX6    X1                                                    COMCIDP    6617    A
COMCIDP          BX7    X2+X7       SET TO *LEN SPECIFIED*                    COMCIDP    6618    A
COMCIDP          SA6    PAT#LEN                                               COMCIDP    6619    A
COMCIDP          SA7    A2+                                                   COMCIDP    6620    A
COMCIDP                                                                       COMCIDP    6621    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *         HERE TO RESOLVE ANY POSSIBLE SYNTACTIC DIFFERENCES                COMCIDP    6622    A
COMCIDP  *         BETWEEN *FWA,LWA* AND *FWA,,LEN*.                                 COMCIDP    6623    A
COMCIDP                                                                              COMCIDP    6624    A
COMCIDP   PAT4     SA1    PATFLG                                                     COMCIDP    6625    A
COMCIDP            SA2    PAT#FWA     (X2) = FWA                                     COMCIDP    6626    A
COMCIDP            SA3    A2+B1       (X3) = LWA                                     COMCIDP    6627    A
COMCIDP            SA4    A3+B1       (X4) = LEN                                     COMCIDP    6628    A
COMCIDP            ZR     X1,PAT6     IF NO PROBLEM                                  COMCIDP    6629    A
COMCIDP            SX6    X1-PAF.LWAM-PAF.LENM                                       COMCIDP    6630    A
COMCIDP            SX7    X1-PAF.LWAM                                                COMCIDP    6631    A
COMCIDP            ZR     X6,PAT6     IF *FWA,LWA,LEN*                               COMCIDP    6632    A
COMCIDP            NZ     X7,PAT5     IF *FWA,,LEN*                                  COMCIDP    6633    A
COMCIDP                                                                              COMCIDP    6634    A
COMCIDP  *         HERE IF *FWA,LWA*.                                                COMCIDP    6635    A
COMCIDP  *         SET: LEN=LWA-FWA+1                                                COMCIDP    6636    A
COMCIDP  *                                                                           COMCIDP    6637    A
COMCIDP  *         NOTE THAT A SPECIAL DEAL HAPPENS HERE TO HELP CLUMSY TYPISTS      COMCIDP    6638    A
COMCIDP  *         AND AN EVEN CLUMSIER EXTERNAL SYNTAX.  IF --                      COMCIDP    6639    A
COMCIDP  *                                                                           COMCIDP    6640    A
COMCIDP  *                LWA .LT. FWA .AND. LWA .LT. 100                            COMCIDP    6641    A
COMCIDP  *                                                                           COMCIDP    6642    A
COMCIDP  *         THEN WE USE *LWA* AS A *LEN*.                                     COMCIDP    6643    A
COMCIDP                                                                              COMCIDP    6644    A
COMCIDP            IX6    X3-X2       (X6) = LWA-FWA                                 COMCIDP    6645    A
COMCIDP            SX4    X6+B1                                                      COMCIDP    6646    A
COMCIDP            PL     X6,PAT6     IF FWA .LE. LWA                                COMCIDP    6647    A
COMCIDP            SX7    X3-100                                                     COMCIDP    6648    A
COMCIDP            PL     X7,PAT6     IF DON'T WANT TO RISK IT                       COMCIDP    6649    A
COMCIDP            BX4    X3          (X4) = LEN                                     COMCIDP    6650    A
COMCIDP                                                                              COMCIDP    6651    A
COMCIDP  *         HERE IF *FWA,,LEN*.                                               COMCIDP    6652    A
COMCIDP  *         SET: LWA = FWA+LEN-1                                              COMCIDP    6653    A
COMCIDP                                                                              COMCIDP    6654    A
COMCIDP   PAT5     IX6    X2+X4                                                      COMCIDP    6655    A
COMCIDP            SX3    X6-1                                                       COMCIDP    6656    A
COMCIDP            BX6    X3                                                         COMCIDP    6657    A
COMCIDP            SA6    A3                                                         COMCIDP    6658    A
COMCIDP                                                                              COMCIDP    6659    A
COMCIDP  *         CHECK *FWA,LWA,LEN*.                                              COMCIDP    6660    A
COMCIDP  *                                                                           COMCIDP    6661    A
COMCIDP  *         ENTRY  (X2) = FWA                                                 COMCIDP    6662    A
COMCIDP  *                (X3) = LWA                                                 COMCIDP    6663    A
COMCIDP  *                (X4) = LEN                                                 COMCIDP    6664    A
COMCIDP                                                                              COMCIDP    6665    A
COMCIDP   PAT6     SX6    X2                                                         COMCIDP    6666    A
COMCIDP            SX7    X3                                                         COMCIDP    6667    A
COMCIDP            SA6    PAT#FWA                                                    COMCIDP    6668    A
COMCIDP            SA7    A6+B1                                                      COMCIDP    6669    A
COMCIDP            SX6    X4                                                         COMCIDP    6670    A
COMCIDP            SA6    A7+1                                                       COMCIDP    6671    A
COMCIDP                                                                              COMCIDP    6672    A
COMCIDP            RJ     FLL         CHECK FWA,LWA,LEN                              COMCIDP    6673    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            PL     B2,PAT7     IF FWA, LWA, AND LEN OK                    COMCIDP    6674    A
COMCIDP            BX2    X2-X2       CLEAR *FWA*                                COMCIDP    6675    A
COMCIDP            MX3    0           CLEAR *LWA*                                COMCIDP    6676    A
COMCIDP            SX4    B0          CLEAR *LEN*                                COMCIDP    6677    A
COMCIDP            EQ     EXIT.                                                  COMCIDP    6678    A
COMCIDP                                                                           COMCIDP    6679    A
COMCIDP  *         SET UP EXIT CONDITIONS.                                       COMCIDP    6680    A
COMCIDP                                                                           COMCIDP    6681    A
COMCIDP   PAT7     SA2    PAT#FWA     (X2) = FWA                                 COMCIDP    6682    A
COMCIDP            SA3    A2+B1       (X3) = LWA                                 COMCIDP    6683    A
COMCIDP            SA4    A3+B1       (X4) = LEN                                 COMCIDP    6684    A
COMCIDP            BX1    X1-X1       SET TO *USE PTR ORD AT (TB=KEY)*           COMCIDP    6685    A
COMCIDP            SB2    B1          SET TO *1ST DEFAULT*                       COMCIDP    6686    A
COMCIDP            SB3    B1+         SET TO *PUT NEW VALUE*                     COMCIDP    6687    A
COMCIDP            RJ     PTR         PUT POINTER VALUES                         COMCIDP    6688    A
COMCIDP            EQ     EXIT.                                                  COMCIDP    6689    A
COMCIDP                                                                           COMCIDP    6690    A
COMCIDP                                                                           COMCIDP    6691    A
COMCIDP            LOC    AP=LL                                                  COMCIDP    6692    A
COMCIDP   PAT#FWA  =      *                                                      COMCIDP    6693    A
COMCIDP   PAT#LWA  =      *+1                                                    COMCIDP    6694    A
COMCIDP   PAT#LEN  =      *+2                                                    COMCIDP    6695    A
COMCIDP            LOC    *O                                                     COMCIDP    6696    A
COMCIDP   PEM      SPACE  4,10                                                   COMCIDP    6697    A
COMCIDP  **        PEM - PRINT ERROR MSG.                                        COMCIDP    6698    A
COMCIDP  *                                                                       COMCIDP    6699    A
COMCIDP  *                                                                       COMCIDP    6700    A
COMCIDP  *         THIS ROUTINE PRINTS AN IDP ERROR MSG IN ONE OF TWO            COMCIDP    6701    A
COMCIDP  *         WAYS --                                                       COMCIDP    6702    A
COMCIDP  *                                                                       COMCIDP    6703    A
COMCIDP  *           1. IF *OI.ERR = 1*, THEN THE ERROR MESSAGE NR IS PRINTED.   COMCIDP    6704    A
COMCIDP  *                                                                       COMCIDP    6705    A
COMCIDP  *           2. IF *OI.ERR = 2*, THEN THE ERROR MESSAGE TEXT IS PRINTED. COMCIDP    6706    A
COMCIDP  *                                                                       COMCIDP    6707    A
COMCIDP  *         SEE *OI.ERR* (IN *COMSIDP*) FOR MORE INFORMATION.             COMCIDP    6708    A
COMCIDP  *                                                                       COMCIDP    6709    A
COMCIDP  *         ENTRY  (IDPERR) = .ZR. IF NO ERROR HAS OCCURRED, ELSE         COMCIDP    6710    A
COMCIDP  *                         = .MI. IF NO ERR MSG AVAILABLE FOR THIS ERROR. COMCIDP    6711    A
COMCIDP  *                         = .PL. IF AN ERROR MSG IS AVAILABLE --        COMCIDP    6712    A
COMCIDP  *                                                                       COMCIDP    6713    A
COMCIDP  *                           1. IF *OI.ERR = 1*, THEN (IDPERR) = THE ERR COMCIDP    6714    A
COMCIDP  *                              MSG NUMBER.                              COMCIDP    6715    A
COMCIDP  *                                                                       COMCIDP    6716    A
COMCIDP  *                           2. IF *OI.ERR = 2*, THEN (IDPERR) = THE FWA COMCIDP    6717    A
COMCIDP  *                              OF THE ERR MSG TO BE PRINTED.            COMCIDP    6718    A
COMCIDP  *                                                                       COMCIDP    6719    A
COMCIDP  *         EXIT   NONE                                                   COMCIDP    6720    A
COMCIDP  *                                                                       COMCIDP    6721    A
COMCIDP  *         USES   ALL                                                    COMCIDP    6722    A
COMCIDP  *                                                                       COMCIDP    6723    A
COMCIDP  *         CALLS  WRITEC                                                 COMCIDP    6724    A
COMCIDP                                                                          COMCIDP    6725    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP                                                                                COMCIDP     6726     A
COMCIDP   PEM      SUBR                  ** ENTRY/EXIT **                              COMCIDP     6727     A
COMCIDP                                                                                COMCIDP     6728     A
COMCIDP   *        CHECK FOR ERR MSG AVAILABILITY.                                     COMCIDP     6729     A
COMCIDP                                                                                COMCIDP     6730     A
COMCIDP            SA1    IDPERR      (X1) = ERR MSG NR IF *OI.ERR = 1*, ELSE          COMCIDP     6731     A
COMCIDP   *                                = FWA OF ERR MSG TEXT IF *OI.ERR = 2*       COMCIDP     6732     A
COMCIDP            NZ     X1,PEM2     IF THERE IS AN ERROR                              COMCIDP     6733     A
COMCIDP            WRITEC F.IDO,PEMA                                                    COMCIDP     6734     A
COMCIDP            EQ     EXIT.                                                         COMCIDP     6735     A
COMCIDP                                                                                COMCIDP     6736     A
COMCIDP   PEM2     PL     X1,PEM3     IF THERE IS AN ERR MSG FOR THIS ERROR             COMCIDP     6737     A
COMCIDP            WRITEC F.IDO,PEMB                                                    COMCIDP     6738     A
COMCIDP            EQ     EXIT.                                                         COMCIDP     6739     A
COMCIDP                                                                                COMCIDP     6740     A
COMCIDP   PEM3     BSS    0                                                             COMCIDP     6741     A
COMCIDP                                                                                COMCIDP     6742     A
COMCIDP   '?ID#001 IFEQ   OI.ERR,1                                                      COMCIDP     6743     A
COMCIDP                                                                                COMCIDP     6744     A
COMCIDP            SA2    =10H ERROR NR                                                 COMCIDP     6745     A
COMCIDP            BX6    X2                                                            COMCIDP     6746     A
COMCIDP            SA6    SNAPLNE                                                       COMCIDP     6747     A
COMCIDP            RJ     =XCDD       CONVERT BINARY TO DECIMAL DPC                     COMCIDP     6748     A
COMCIDP            BX7    X7-X7                                                         COMCIDP     6749     A
COMCIDP            SA6    A6+B1                                                         COMCIDP     6750     A
COMCIDP            SA7    A7+B1       MARK EOL                                          COMCIDP     6751     A
COMCIDP            WRITEC =XF.IDO,SNAPLNE                                               COMCIDP     6752     A
COMCIDP            EQ     EXIT.                                                         COMCIDP     6753     A
COMCIDP                                                                                COMCIDP     6754     A
COMCIDP   '?ID#001 ELSE                                                                 COMCIDP     6755     A
COMCIDP                                                                                COMCIDP     6756     A
COMCIDP            WRITEC =XF.IDO,X1                                                    COMCIDP     6757     A
COMCIDP            EQ     EXIT.                                                         COMCIDP     6758     A
COMCIDP   '?ID#001 ENDIF                                                                COMCIDP     6759     A
COMCIDP                                                                                COMCIDP     6760     A
COMCIDP                                                                                COMCIDP     6761     A
COMCIDP   PEMA     DATA   C. NO ERROR.                                                  COMCIDP     6762     A
COMCIDP                                                                                COMCIDP     6763     A
COMCIDP   PEMB     DATA   C. NO ERROR MESSAGE FOR THIS ERROR.                           COMCIDP     6764     A
COMCIDP   POL      SPACE  4,10                                                          COMCIDP     6765     A
COMCIDP   **       POL - PROCESS OPTIONS LIST ITEM.                                     COMCIDP     6766     A
COMCIDP   *                                                                             COMCIDP     6767     A
COMCIDP   *                                                                             COMCIDP     6768     A
COMCIDP   *               THIS ROUTINE WILL PROCESS AN OPTIONS LIST ITEM THAT IS        COMCIDP     6769     A
COMCIDP   *        TERMINATED BY A -,- OR -EOS-.                                        COMCIDP     6770     A
COMCIDP   *                                                                             COMCIDP     6771     A
COMCIDP   *        ENTRY  (B6) = FWA OF OPTIONS KEYWORD TABLE (OKT)                     COMCIDP     6772     A
COMCIDP   *               (A5+1,X5) = ADDR + CONTENTS OF 1ST TOKEN IN LIST              COMCIDP     6773     A
COMCIDP   *                                                                             COMCIDP     6774     A
COMCIDP   *        EXIT   IF NO ERROR--                                                 COMCIDP     6775     A
COMCIDP   *               (A5,X5) =  A + C OF TOKEN THAT TERMINATED OPTIONS LIST        COMCIDP     6776     A
COMCIDP   *               (X3)    = .ZR. IF OPTIONS LIST ITEM WAS NULL, ELSE .NZ.      COMCIDP     6777     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *                (X6)    =  -DEFAULT IF - PREFIX, ELSE +DEFAULT                COMCIDP    6778    A
COMCIDP  *                (X7)    =  +0 IF - PREFIX, ELSE + DEFAULT                     COMCIDP    6779    A
COMCIDP  *                (B7)    =  ORDINAL INTO *OKT* OF FIND                         COMCIDP    6780    A
COMCIDP  *                                                                              COMCIDP    6781    A
COMCIDP  *                IF AN ERROR-- EXIT IS TO *IDP=ER*                             COMCIDP    6782    A
COMCIDP  *                                                                              COMCIDP    6783    A
COMCIDP  *         USES   X - ALL                                                       COMCIDP    6784    A
COMCIDP  *                A - 2,3,5                                                     COMCIDP    6785    A
COMCIDP  *                B - 7                                                         COMCIDP    6786    A
COMCIDP  *                                                                              COMCIDP    6787    A
COMCIDP  *         CALLS  SKT                                                           COMCIDP    6788    A
COMCIDP                                                                                 COMCIDP    6789    A
COMCIDP                                                                                 COMCIDP    6790    A
COMCIDP   POL      SUBR                ** ENTRY/EXIT **                                 COMCIDP    6791    A
COMCIDP                                                                                 COMCIDP    6792    A
COMCIDP  *         INITIALIZATION.                                                      COMCIDP    6793    A
COMCIDP                                                                                 COMCIDP    6794    A
COMCIDP            BX4    X4-X4       SET TO *NO - PREFIX SO FAR*                       COMCIDP    6795    A
COMCIDP            MX6    -1                                                            COMCIDP    6796    A
COMCIDP            SX7    B0                                                            COMCIDP    6797    A
COMCIDP            SA6    IDPNUL      SET TO *NULL EXPRESSION SO FAR*                   COMCIDP    6798    A
COMCIDP            SA7    IDPACC      CLEAR ACCUMULATOR                                 COMCIDP    6799    A
COMCIDP                                                                                 COMCIDP    6800    A
COMCIDP  *         MAIN LOOP NODE.                                                      COMCIDP    6801    A
COMCIDP                                                                                 COMCIDP    6802    A
COMCIDP   POL2     SA3    IDPNUL                                                        COMCIDP    6803    A
COMCIDP            SA5    A5+B1                                                         COMCIDP    6804    A
COMCIDP            SX6    X3+B1                                                         COMCIDP    6805    A
COMCIDP            MX0    -CHAR                                                         COMCIDP    6806    A
COMCIDP            SA6    A3                                                            COMCIDP    6807    A
COMCIDP            ZR     X5,POL4     IF EOS ENCOUNTERED                                COMCIDP    6808    A
COMCIDP            SX6    X5-O.VAR                                                      COMCIDP    6809    A
COMCIDP            SX7    X5-O.COM                                                      COMCIDP    6810    A
COMCIDP            ZR     X6,POL3     IF *VAR* TOKEN                                    COMCIDP    6811    A
COMCIDP            ZR     X7,POL4     IF , TOKEN                                        COMCIDP    6812    A
COMCIDP            SX6    X5-O.MIN                                                      COMCIDP    6813    A
COMCIDP            NZ     X6,E.OPT    IF NOT - TOKEN, ERROR...                          COMCIDP    6814    A
COMCIDP            MX4    60          SET TO * - PREFIX ENCOUNTERED*                    COMCIDP    6815    A
COMCIDP            EQ     POL2                                                          COMCIDP    6816    A
COMCIDP                                                                                 COMCIDP    6817    A
COMCIDP  *         HERE TO PROCESS NAME TOKEN.                                          COMCIDP    6818    A
COMCIDP                                                                                 COMCIDP    6819    A
COMCIDP   POL3     MX0    7*CHAR                                                        COMCIDP    6820    A
COMCIDP            SX6    B1                                                            COMCIDP    6821    A
COMCIDP            SA2    =40404040404040404040B                                        COMCIDP    6822    A
COMCIDP            BX1    X0*X5       (X1) = 0LKEYWORD                                  COMCIDP    6823    A
COMCIDP            IX6    X1-X6       BORROW RIPPLES LEFT TO 1ST NON-ZERO BIT           COMCIDP    6824    A
COMCIDP            BX7    -X6+X1      (X7) = ALL TRAILING 0 BITS IN (X1) = 0 BITS       COMCIDP    6825    A
COMCIDP  *                                   ALL OTHER LEADING BITS = 1 BITS            COMCIDP    6826    A
COMCIDP            SB7    60-5        (B7) = RIGHT CIRCULAR SHIFT 5 BITS                COMCIDP    6827    A
COMCIDP            BX6    X2*X7       (X6) = 40B WHERE EACH CHAR IN KEYWORD IS          COMCIDP    6828    A
COMCIDP            LX7    X6,B7       (X7) = 01B WHERE EACH CHAR IN KEYWORD IS          COMCIDP    6829    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            IX7    X6-X7       (X7) = 37B WHERE EACH CHAR IN KEYWORD IS          COMCIDP    6830    A
COMCIDP            BX0    X6+X7       (X0) = 77B WHERE EACH CHAR IN KEYWORD IS          COMCIDP    6831    A
COMCIDP            RJ     SKT         SEARCH FOR NAME                                   COMCIDP    6832    A
COMCIDP            ZR     X2,E.OPT    IF NO FIND                                        COMCIDP    6833    A
COMCIDP            SX6    X2                                                            COMCIDP    6834    A
COMCIDP            SA6    IDPACC                                                        COMCIDP    6835    A
COMCIDP            EQ     POL2                                                          COMCIDP    6836    A
COMCIDP                                                                                 COMCIDP    6837    A
COMCIDP  *         HERE TO SET UP EXIT CONDITIONS.                                      COMCIDP    6838    A
COMCIDP                                                                                 COMCIDP    6839    A
COMCIDP   POL4     SA1    IDPACC                                                        COMCIDP    6840    A
COMCIDP            SA3    IDPNUL                                                        COMCIDP    6841    A
COMCIDP            BX6    X1-X4       (X6) = -DEFAULT IF - PREFIX, ELSE +DEFAULT        COMCIDP    6842    A
COMCIDP            BX7    -X4*X1      (X7) = +0 IF - PREFIX, ELSE +DEFAULT              COMCIDP    6843    A
COMCIDP            EQ     EXIT.                                                         COMCIDP    6844    A
COMCIDP   PTR      SPACE  4,10                                                          COMCIDP    6845    A
COMCIDP  **        PTR - POINTER MANAGER.                                               COMCIDP    6846    A
COMCIDP  *                                                                              COMCIDP    6847    A
COMCIDP  *                                                                              COMCIDP    6848    A
COMCIDP  *         *PTR* PERFORMS THE TASK OF MANAGING THE ACCESSING                    COMCIDP    6849    A
COMCIDP  *         AND CHANGING OF IDP'S USER POINTERS (AT *FW.PTR* ET SEQ).            COMCIDP    6850    A
COMCIDP  *                                                                              COMCIDP    6851    A
COMCIDP  *         ENTRY  (X1) = .NZ. IF A *KEYW* ENTRY CONTAINING *PTR* ORD            COMCIDP    6852    A
COMCIDP  *                       TO USE (SEE *KEYW* MACRO AND *KW.XXXP* AND             COMCIDP    6853    A
COMCIDP  *                       *KW.XXXL* SYMBOL DEFINITIONS IN *COMSIDP*).            COMCIDP    6854    A
COMCIDP  *                     = .ZR. IF *PTR* IS TO USE THE *KEYW* ENTRY               COMCIDP    6855    A
COMCIDP  *                       AT (TB=KEY).                                           COMCIDP    6856    A
COMCIDP  *                (B3) = .PL. IF PUTTING/STORING NEW *PTR* VALUE.               COMCIDP    6857    A
COMCIDP  *                     = .MI. IF GETTING OLD/CURRENT *PTR* VALUE.               COMCIDP    6858    A
COMCIDP  *                (B2) = *PTR* LEVEL THAT CALLER IS PUTTING/GETTING.            COMCIDP    6859    A
COMCIDP  *                       I.E. 1ST DEFAULT IS (B2) = 1, 2ND DEFAULT              COMCIDP    6860    A
COMCIDP  *                       IS (B2) = 2, ETC.  NOTE THAT *PTR* CHECKS              COMCIDP    6861    A
COMCIDP  *                       TO MAKE SURE CALLER IS REQUESTING A LEGAL              COMCIDP    6862    A
COMCIDP  *                       (I.E. EXISTENT) *PTR* LEVEL.                           COMCIDP    6863    A
COMCIDP  *                                                                              COMCIDP    6864    A
COMCIDP  *                IF PUTTING NEW VALUE, (B3) .PL. --                            COMCIDP    6865    A
COMCIDP  *                                                                              COMCIDP    6866    A
COMCIDP  *                (X2) = FWA.                                                   COMCIDP    6867    A
COMCIDP  *                (X4) = LEN.  NOTE THAT (X3) IS IGNORED.                       COMCIDP    6868    A
COMCIDP  *                                                                              COMCIDP    6869    A
COMCIDP  *         EXIT   (X2) = FWA.                                                   COMCIDP    6870    A
COMCIDP  *                (X4) = LEN.                                                   COMCIDP    6871    A
COMCIDP  *                (B2) = ADDR OF *PTR* ENTRY PUTTEN/GOTTEN.                     COMCIDP    6872    A
COMCIDP  *                                                                              COMCIDP    6873    A
COMCIDP  *         USES   X - 1,2,4,6,7                                                 COMCIDP    6874    A
COMCIDP  *                A - 1,6                                                       COMCIDP    6875    A
COMCIDP  *                B - 2,3,4                                                     COMCIDP    6876    A
COMCIDP  *                                                                              COMCIDP    6877    A
COMCIDP  *         CALLS  NONE                                                          COMCIDP    6878    A
COMCIDP                                                                                 COMCIDP    6879    A
COMCIDP                                                                                 COMCIDP    6880    A
COMCIDP   PTR      SUBR                ** ENTRY/EXIT **                                 COMCIDP    6881    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            SB4    L1.PTR      (B4) = NR OF DEFAULTS FOR EACH *PTR*          COMCIDP    6882    A
COMCIDP            NZ     X1,PTR2     IF CALLER PROVIDED POINTER DEF KEYW           COMCIDP    6883    A
COMCIDP            SA1    TB=KEY                                                    COMCIDP    6884    A
COMCIDP            SA1    X1+         (X1) = *KEYW* ENTRY                           COMCIDP    6885    A
COMCIDP                                                                             COMCIDP    6886    A
COMCIDP   PTR2     MX7    -KW.PNRL                                                  COMCIDP    6887    A
COMCIDP            LX1    0-KW.PNRP                                                 COMCIDP    6888    A
COMCIDP            BX6    -X7*X1      (X6) = *PTR* ORD                              COMCIDP    6889    A
COMCIDP            SX7    B4                                                        COMCIDP    6890    A
COMCIDP            GT     B2,B4,SE.BPTR IF REQUESTING NON-EXISTENT DEFAULT PTR      COMCIDP    6891    A
COMCIDP            LE     B2,B0,SE.BPTR IF REQUESTING NON-EXISTENT DEFAULT PTR      COMCIDP    6892    A
COMCIDP            IX1    X6*X7       (X1) = ORD INTO PTR TBL + L1.PTR              COMCIDP    6893    A
COMCIDP            NO                                                                COMCIDP    6894    A
COMCIDP            SX6    X1-L.PTR-L1.PTR                                           COMCIDP    6895    A
COMCIDP            PL     X6,SE.BPTR  IF OVER-INDEXING PTR TABLE                    COMCIDP    6896    A
COMCIDP            SB4    FW.PTR+X1-1                                               COMCIDP    6897    A
COMCIDP            SB2    B4+B2                                                     COMCIDP    6898    A
COMCIDP            PL     B3,PTR3     IF PUTTING NEW VALUES                         COMCIDP    6899    A
COMCIDP                                                                             COMCIDP    6900    A
COMCIDP  *         HERE IF GETTING CURRENT VALUES.                                  COMCIDP    6901    A
COMCIDP                                                                             COMCIDP    6902    A
COMCIDP            SA1    B2          (A1,X1) = A+C OF REQUESTED *PTR*              COMCIDP    6903    A
COMCIDP            LX1    0-PT.FWAP                                                 COMCIDP    6904    A
COMCIDP            SX2    X1          (X2) = FWA                                    COMCIDP    6905    A
COMCIDP            LX1    0-PT.LENP+PT.FWAP-0                                       COMCIDP    6906    A
COMCIDP            SX4    X1+         (X4) = LEN                                    COMCIDP    6907    A
COMCIDP            EQ     EXIT.                                                     COMCIDP    6908    A
COMCIDP                                                                             COMCIDP    6909    A
COMCIDP  *         HERE IF PUTTING NEW VALUES.                                      COMCIDP    6910    A
COMCIDP                                                                             COMCIDP    6911    A
COMCIDP   PTR3     SX6    X2                                                        COMCIDP    6912    A
COMCIDP            SX7    X4                                                        COMCIDP    6913    A
COMCIDP            LX6    PT.FWAP-0                                                 COMCIDP    6914    A
COMCIDP            LX7    PT.LENP-0                                                 COMCIDP    6915    A
COMCIDP            BX6    X6+X7                                                     COMCIDP    6916    A
COMCIDP            SA6    B2          STORE NEW *PTR*                               COMCIDP    6917    A
COMCIDP            EQ     EXIT.                                                     COMCIDP    6918    A
COMCIDP   RIL      SPACE  4,10                                                      COMCIDP    6919    A
COMCIDP  **        RIL - READ IDP INPUT LINE.                                       COMCIDP    6920    A
COMCIDP  *                                                                          COMCIDP    6921    A
COMCIDP  *                                                                          COMCIDP    6922    A
COMCIDP  *         ENTRY  APPROPRIATE *IDPFLG* CONTROL BITS SET.                    COMCIDP    6923    A
COMCIDP  *                                                                          COMCIDP    6924    A
COMCIDP  *         EXIT   (X1)      = .NZ. IF EOR/EOF ENCOUNTERED, ELSE .ZR.        COMCIDP    6925    A
COMCIDP  *                IF NO EOR/EOF ENCOUNTERED --                             COMCIDP    6926    A
COMCIDP  *                (L=LIM)   = NR OF WORDS IN SOURCE LINE IMAGE               COMCIDP    6927    A
COMCIDP  *                            (INCLUDES WORD CONTAINING EOL MARK)            COMCIDP    6928    A
COMCIDP  *                (TC=SOB)  = NR OF BITS IN FIRST WORD OF SOURCE LINE,     COMCIDP    6929    A
COMCIDP  *                            (FOR *COMCTOK*).                               COMCIDP    6930    A
COMCIDP  *                (TC=SOL)  = NR OF BITS IN SOURCE LINE IMAGE,             COMCIDP    6931    A
COMCIDP  *                            (FOR *COMCTOK*).                               COMCIDP    6932    A
COMCIDP  *                                                                          COMCIDP    6933    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *         USES   ALL BUT A0,X0     (INCLUDES ALL CALLS)                COMCIDP     6934     A
COMCIDP  *                                                                      COMCIDP     6935     A
COMCIDP  *         CALLS  READ,READC,SFN                                        COMCIDP     6936     A
COMCIDP                                                                         COMCIDP     6937     A
COMCIDP                                                                         COMCIDP     6938     A
COMCIDP   RIL      SUBR                ** ENTRY/EXIT **                         COMCIDP     6939     A
COMCIDP                                                                         COMCIDP     6940     A
COMCIDP            SA5    IDPFLG                                                COMCIDP     6941     A
COMCIDP            SX2    F.IDI                                                 COMCIDP     6942     A
COMCIDP            LX5    59-IDF.INPP                                           COMCIDP     6943     A
COMCIDP            SB6    FW.LIM                                                COMCIDP     6944     A
COMCIDP            PL     X5,RIL2     IF READING INTERACTIVELY (F.IDI)          COMCIDP     6945     A
COMCIDP            SX2    F.BDI                                                 COMCIDP     6946     A
COMCIDP                                                                         COMCIDP     6947     A
COMCIDP   RIL2     LX5    59-IDF.RAPP+IDF.INPP-59                               COMCIDP     6948     A
COMCIDP            MI     X5,RIL3     IF READ ALREADY PERFORMED                 COMCIDP     6949     A
COMCIDP            READ   X2                                                    COMCIDP     6950     A
COMCIDP                                                                         COMCIDP     6951     A
COMCIDP  *         SET READ ALREADY PERFORMED (RAP) STATUS.                     COMCIDP     6952     A
COMCIDP                                                                         COMCIDP     6953     A
COMCIDP            MX6    1                                                     COMCIDP     6954     A
COMCIDP            BX6    X6+X5       SET RAP                                   COMCIDP     6955     A
COMCIDP            LX6    IDF.RAPP-59                                           COMCIDP     6956     A
COMCIDP            SA6    A5                                                    COMCIDP     6957     A
COMCIDP                                                                         COMCIDP     6958     A
COMCIDP   RIL3     READC  X2,B6,8                                               COMCIDP     6959     A
COMCIDP            NZ     X1,EXIT.    IF EOR/EOF STATUS                         COMCIDP     6960     A
COMCIDP                                                                         COMCIDP     6961     A
COMCIDP  *         SET UP EXIT CONDITIONS.                                      COMCIDP     6962     A
COMCIDP                                                                         COMCIDP     6963     A
COMCIDP   RIL4     SX6    B6-FW.LIM   (X6) = NR OF WORDS IN SOURCE LINE         COMCIDP     6964     A
COMCIDP            SA6    L=LIM                                                 COMCIDP     6965     A
COMCIDP            SB7    X6+                                                   COMCIDP     6966     A
COMCIDP            LE     B7,B1,RIL5  IF 11 CHAR (66 BIT) EOL NOT POSSIBLE      COMCIDP     6967     A
COMCIDP            SA1    FW.LIM+B7-2 (X1) = NEXT TO LAST WORD OF LINE          COMCIDP     6968     A
COMCIDP            MX2    -CHAR                                                 COMCIDP     6969     A
COMCIDP            BX3    -X2*X1                                                COMCIDP     6970     A
COMCIDP            NZ     X3,RIL5     IF NOT 11 CHAR (66 BIT) EOL MARK          COMCIDP     6971     A
COMCIDP            SB7    B7-1                                                  COMCIDP     6972     A
COMCIDP                                                                         COMCIDP     6973     A
COMCIDP   RIL5     SA1    FW.LIM+B7-1 (X1) = LAST WORD OF SOURCE LINE           COMCIDP     6974     A
COMCIDP            RJ     =XSFN       SPACE FILL NAME                           COMCIDP     6975     A
COMCIDP            CX6    X7          (X6) = NR OF BITS IN LAST WORD            COMCIDP     6976     A
COMCIDP            SX7    X6          (X7) = NR OF BITS IN 1ST WORD IF 1ST = LAST COMCIDP     6977     A
COMCIDP            LE     B7,B1,RIL6  IF 1ST WORD IS ALSO LAST WORD             COMCIDP     6978     A
COMCIDP            SB7    B7-1                                                  COMCIDP     6979     A
COMCIDP            SX7    B7                                                    COMCIDP     6980     A
COMCIDP            SB7    B7+B7       *2                                        COMCIDP     6981     A
COMCIDP            LX7    6           *64                                       COMCIDP     6982     A
COMCIDP            SX2    B7+B7       *4                                        COMCIDP     6983     A
COMCIDP            IX7    X7-X2       *60                                       COMCIDP     6984     A
COMCIDP            IX6    X7+X6       (X6) = NR OF BITS IN SOURCE LINE          COMCIDP     6985     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            SX7    10*CHAR     (X7) = NR OF BITS IN 1ST WORD                    COMCIDP     6986     A
COMCIDP                                                                                COMCIDP     6987     A
COMCIDP   RIL6     SX1    0           SET TO *NO EOR/EOF/EOI ENCOUNTERED*              COMCIDP     6988     A
COMCIDP            SA6    TC=SOL                                                       COMCIDP     6989     A
COMCIDP            SA7    TC=SOB                                                       COMCIDP     6990     A
COMCIDP            EQ     EXIT.                                                        COMCIDP     6991     A
COMCIDP   .IDPOS   ENDIF                                                               CCGA018       29     A
COMCIDP   RIR      SPACE  4,10                                                         COMCIDP     6992     A
COMCIDP  **        RIR - RESTORE INTERACTIVE REGISTERS.                                COMCIDP     6993     A
COMCIDP  *                                                                             COMCIDP     6994     A
COMCIDP  *                                                                             COMCIDP     6995     A
COMCIDP  *         *RIR* RESTORES ALL THE IDP CALLER'S REGISTERS FROM THE              COMCIDP     6996     A
COMCIDP  *         REGISTER SAVE AREA PROVIDE ON ENTRY IN (X1).                        COMCIDP     6997     A
COMCIDP  *                                                                             COMCIDP     6998     A
COMCIDP  *         *RIR* MAKES THE DYNAMIC DECISION AS TO WHETHER TO CALL              COMCIDP     6999     A
COMCIDP  *         *COMCRSR* OR *COMCXJR* TO RESTORE REGISTERS, BASED ON               COMCIDP     7000     A
COMCIDP  *         WHETHER THE IDP INTERACTIVE USER HAS SELECTED --                    COMCIDP     7001     A
COMCIDP  *                                                                             COMCIDP     7002     A
COMCIDP  *                OPTION,XJR                                                   COMCIDP     7003     A
COMCIDP  *                                                                             COMCIDP     7004     A
COMCIDP  *         OR NOT.                                                             COMCIDP     7005     A
COMCIDP  *                                                                             COMCIDP     7006     A
COMCIDP  *         ENTRY  (X1) = FWA OF REGISTER SAVE AREA                             COMCIDP     7007     A
COMCIDP  *                                                                             COMCIDP     7008     A
COMCIDP  *         EXIT   ALL REGISTERS RESTORED TO CONTENTS OF REGISTER               COMCIDP     7009     A
COMCIDP  *                SAVE AREA.                                                   COMCIDP     7010     A
COMCIDP  *                                                                             COMCIDP     7011     A
COMCIDP  *         USES   ALL                                                          COMCIDP     7012     A
COMCIDP  *                                                                             COMCIDP     7013     A
COMCIDP  *         CALLS  RSR,XJR                                                      COMCIDP     7014     A
COMCIDP                                                                                COMCIDP     7015     A
COMCIDP                                                                                COMCIDP     7016     A
COMCIDP   RIR      SUBR              ** ENTRY/EXIT **                                  COMCIDP     7017     A
COMCIDP            SA2    IDPFLG                                                       COMCIDP     7018     A
COMCIDP            LX2    59-IDF.XJRP                                                  COMCIDP     7019     A
COMCIDP            MI     X2,RIR2     IF *OPT,XJR* SELECTED                            COMCIDP     7020     A
COMCIDP                                                                                COMCIDP     7021     A
COMCIDP            RJ     =XRSR       RESTORE REGISTERS VIA CPU                        COMCIDP     7022     A
COMCIDP            EQ     EXIT.                                                        COMCIDP     7023     A
COMCIDP                                                                                COMCIDP     7024     A
COMCIDP   RIR2     BSS    0                                                            COMCIDP     7025     A
COMCIDP            RJ     =XXJR       RESTORE REGISTERS VIA XJR                        COMCIDP     7026          I
          -CCGA018
COMCIDP   .IDPOS   IFNE   .OS,2                                                        CCGA018       30     A
COMCIDP            RJ     =XXJR       RESTORE REGISTERS VIA *XJR*                      CCGA018       31     A
COMCIDP   .IDPOS   ELSE                                                                CCGA018       32     A
COMCIDP            EQ     *+4S15      *XJR* DOES NOT EXIST ON SCOPE 2                  CCGA018       33     A
COMCIDP   .IDPOS   ENDIF                                                               CCGA018       34     A
COMCIDP            EQ     EXIT.                                                        COMCIDP     7027     A
COMCIDP   ROL      SPACE  4,10                                                         COMCIDP     7028     A
COMCIDP  **        ROL - WRITE OUTPUT LINE.                                            COMCIDP     7029     A
COMCIDP  *                                                                             COMCIDP     7030     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *                                                                  COMCIDP    7031    A
COMCIDP  *                SHOULD BE ACCESSED VIA *PRIDP* MACRO.  THIS ROUTINE  COMCIDP    7032    A
COMCIDP  *         WILL OUTPUT A SINGLE LINE WITH A SPECIFIED NUMBER OF PRECEDING  COMCIDP    7033    A
COMCIDP  *         BLANK LINES TO EITHER THE INTERACTIVE DEBUG OUTPUT FILE  COMCIDP    7034    A
COMCIDP  *         *F.IDO* AND/OR TO THE USERS BATCH DEBUG OUTPUT FILE  COMCIDP    7035    A
COMCIDP  *         (VIA *PRBDO* MACRO).                                     COMCIDP    7036    A
COMCIDP  *                                                                  COMCIDP    7037    A
COMCIDP  *         ENTRY  (X4) = NR OF PRECEDING BLANK LINES TO OUTPUT (0=NONE)  COMCIDP    7038    A
COMCIDP  *                (X6) = FWA OF LINE TO BE OUTPUT                  COMCIDP    7039    A
COMCIDP  *                (X7) = LEN OF LINE TO BE OUTPUT (IN WORDS)       COMCIDP    7040    A
COMCIDP  *                     = .ZR. IF LEN TO BE COMPUTED BY *WRITEC* (SCOPE 2)  COMCIDP    7041    A
COMCIDP  *                (IDPFLG) --                                      COMCIDP    7042    A
COMCIDP  *                  BIT IDF.IDOP = 1 IF WRITING TO *F.IDO*, ELSE 0  COMCIDP    7043    A
COMCIDP  *                  BIT IDF.BDOP = 1 IF WRITING TO *F.BDO*, ELSE 0  COMCIDP    7044    A
COMCIDP  *                                                                  COMCIDP    7045    A
COMCIDP  *         EXIT   NONE                                              COMCIDP    7046    A
COMCIDP  *                                                                  COMCIDP    7047    A
COMCIDP  *         USES   ALL BUT B4,A0,X0,A5,X5          (INCLUDES ALL CALLS)  COMCIDP    7048    A
COMCIDP  *                                                                  COMCIDP    7049    A
COMCIDP  *         CALLS  PRBDO,WRITEC                                      COMCIDP    7050    A
COMCIDP                                                                     COMCIDP    7051    A
COMCIDP                                                                     COMCIDP    7052    A
COMCIDP   ROL=     SUBR               ** ENTRY/EXIT **                      COMCIDP    7053    A
COMCIDP                                                                     COMCIDP    7054    A
COMCIDP  *         INITIALIZATION.  SAVE CRITICAL REGISTERS TO AVOID LETTING  COMCIDP    7055    A
COMCIDP  *         HOST DESTROY THEM VIA *PRBDO* CALL.                      COMCIDP    7056    A
COMCIDP                                                                     COMCIDP    7057    A
COMCIDP            SA6    ROLA        SAVE (X6) = FWA OF LINE               COMCIDP    7058    A
COMCIDP            SA7    A6+B1       SAVE (X7) = LEN OF LINE               COMCIDP    7059    A
COMCIDP            BX6    X4                                                COMCIDP    7060    A
COMCIDP            SX7    B4-B0                                             COMCIDP    7061    A
COMCIDP            SA6    A7+B1       SAVE (X4) = NR OF PRECEDING BLANK LINES  COMCIDP    7062    A
COMCIDP            SA7    A6+B1       SAVE (B4)                             COMCIDP    7063    A
COMCIDP            SX6    A0-B0                                             COMCIDP    7064    A
COMCIDP            SX7    A5-B0                                             COMCIDP    7065    A
COMCIDP            SA6    A7+B1       SAVE (A0)                             COMCIDP    7066    A
COMCIDP            SA7    A6+B1       SAVE (A5)                             COMCIDP    7067    A
COMCIDP            BX6    X0                                                COMCIDP    7068    A
COMCIDP            LX7    X5                                                COMCIDP    7069    A
COMCIDP            SA6    A7+B1       SAVE (X0)                             COMCIDP    7070    A
COMCIDP            SA7    A6+B1       SAVE (X5)                             COMCIDP    7071    A
COMCIDP                                                                     COMCIDP    7072    A
COMCIDP  *         HERE TO OUTPUT APPROPRIATE NUMBER OF PRECEDING BLANK LINES.  COMCIDP    7073    A
COMCIDP                                                                     COMCIDP    7074    A
COMCIDP   ROL2     SA1    ROL=PRB     (X1) = NR OF PRECEDING BLANK LINES    COMCIDP    7075    A
COMCIDP            SA2    IDPFLG                                            COMCIDP    7076    A
COMCIDP            SX6    X1-1                                              COMCIDP    7077    A
COMCIDP            MI     X6,ROL4     IF NO MORE BLANK LINES                COMCIDP    7078    A
COMCIDP            LX2    59-IDF.IDOP                                       COMCIDP    7079    A
COMCIDP            SA6    A1                                                COMCIDP    7080    A
COMCIDP            PL     X2,ROL3     IF NOT WRITING TO INTERACTIVE DEBUG OUTPUT  COMCIDP    7081    A
COMCIDP                                                                     COMCIDP    7082    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP   #OS2     IFNE   .OS,2       IF NOT SCOPE 2                          COMCIDP     7083     A
COMCIDP            WRITEC =XF.IDO,(=C= =),1                                   COMCIDP     7084     A
COMCIDP                                                                       COMCIDP     7085     A
COMCIDP   #OS2     ELSE                                                       COMCIDP     7086     A
COMCIDP            EQ     *+4S15      IF TRYING TO WRITE INTERACTIVELY...     COMCIDP     7087     A
COMCIDP   #OS2     ENDIF                                                      COMCIDP     7088     A
COMCIDP                                                                       COMCIDP     7089     A
COMCIDP   ROL3     SA1    IDPFLG                                              COMCIDP     7090     A
COMCIDP            LX1    59-IDF.BDOP                                         COMCIDP     7091     A
COMCIDP            PL     X1,ROL2     IF NOT WRITING TO BATCH DEBUG OUTPUT    COMCIDP     7092     A
COMCIDP            PRBDO  (=C= =),1                                           COMCIDP     7093     A
COMCIDP            EQ     ROL2                                                COMCIDP     7094     A
COMCIDP                                                                       COMCIDP     7095     A
COMCIDP  *         HERE TO OUTPUT LINE FINALLY.                               COMCIDP     7096     A
COMCIDP                                                                       COMCIDP     7097     A
COMCIDP   ROL4     SA1    IDPFLG                                              COMCIDP     7098     A
COMCIDP            LX1    59-IDF.IDOP                                         COMCIDP     7099     A
COMCIDP            PL     X1,ROL5     IF NOT WRITING TO INTERACTIVE DEBUG OUTPUT COMCIDP  7100     A
COMCIDP                                                                       COMCIDP     7101     A
COMCIDP   #OS2     IFNE   .OS,2       IF NOT SCOPE 2                          COMCIDP     7102     A
COMCIDP            SA1    ROLA        (X1) = FWA OF LINE                      COMCIDP     7103     A
COMCIDP            SA2    A1+B1       (X2) = LEN OF LINE                      COMCIDP     7104     A
COMCIDP            ZR     X1,ROL6     IF NO LINE TO OUTPUT                    COMCIDP     7105     A
COMCIDP            WRITEC =XF.IDO,X1,X2                                       COMCIDP     7106     A
COMCIDP                                                                       COMCIDP     7107     A
COMCIDP   #OS2     ELSE                                                       COMCIDP     7108     A
COMCIDP            EQ     *+4S15      IF TRYING TO WRITE INTERACTIVELY...     COMCIDP     7109     A
COMCIDP   #OS2     ENDIF                                                      COMCIDP     7110     A
COMCIDP                                                                       COMCIDP     7111     A
COMCIDP   ROL5     SA1    IDPFLG                                              COMCIDP     7112     A
COMCIDP            SA2    ROLA                                                COMCIDP     7113     A
COMCIDP            SA3    A2+B1                                               COMCIDP     7114     A
COMCIDP            LX1    59-IDF.BDOP                                         COMCIDP     7115     A
COMCIDP            PL     X1,ROL6     IF NOT WRITING TO BATCH DEBUG OUTPUT    COMCIDP     7116     A
COMCIDP            ZR     X2,ROL6     IF NO LINE TO OUTPUT                    COMCIDP     7117     A
COMCIDP            PRBDO  X2,X3                                               COMCIDP     7118     A
COMCIDP                                                                       COMCIDP     7119     A
COMCIDP  *         FINAL PROCESSING -- RESTORE SAVED REGISTERS.               COMCIDP     7120     A
COMCIDP                                                                       COMCIDP     7121     A
COMCIDP   ROL6     SA1    ROL=B4                                              COMCIDP     7122     A
COMCIDP            SA2    A1+B1                                               COMCIDP     7123     A
COMCIDP            SA3    A2+B1                                               COMCIDP     7124     A
COMCIDP            SA4    A3+B1                                               COMCIDP     7125     A
COMCIDP            SB4    X1          RESTORE (B4)                            COMCIDP     7126     A
COMCIDP            SA0    X2          RESTORE (A0)                            COMCIDP     7127     A
COMCIDP            SA5    X3          RESTORE (A5)                            COMCIDP     7128     A
COMCIDP            SA1    A4+B1                                               COMCIDP     7129     A
COMCIDP            NO                                                         COMCIDP     7130     A
COMCIDP            BX0    X4          RESTORE (X0)                            COMCIDP     7131     A
COMCIDP            LX5    X1          RESTORE (X5)                            COMCIDP     7132     A
COMCIDP            EQ     EXIT.                                               COMCIDP     7133     A
COMCIDP                                                                       COMCIDP     7134     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP                                                                                  COMCIDP     7135     A
COMCIDP   ROLA     BSSZ   8                                                              COMCIDP     7136     A
COMCIDP                                                                                  COMCIDP     7137     A
COMCIDP            LOC    ROLA                                                           COMCIDP     7138     A
COMCIDP   ROL=FWA  =      *           SAVED FWA                                          COMCIDP     7139     A
COMCIDP   ROL=LEN  =      *+1         SAVED LENGTH                                       COMCIDP     7140     A
COMCIDP   ROL=PRB  =      *+2         SAVED PRECEDING BLANK COUNT                        COMCIDP     7141     A
COMCIDP   ROL=B4   =      *+3         SAVED (B4)                                         COMCIDP     7142     A
COMCIDP   ROL=A0   =      *+4         SAVED (A0)                                         COMCIDP     7143     A
COMCIDP   ROL=A5   =      *+5         SAVED (A5)                                         COMCIDP     7144     A
COMCIDP   ROL=X0   =      *+6         SAVED (X0)                                         COMCIDP     7145     A
COMCIDP   ROL=X5   =      *+7         SAVED (X5)                                         COMCIDP     7146     A
COMCIDP            LOC    *O                                                             COMCIDP     7147     A
COMCIDP   SKT      SPACE  4,10                                                           COMCIDP     7148     A
COMCIDP  **        SKT - SEARCH KEYWORD TABLE.                                           COMCIDP     7149     A
COMCIDP  *                                                                               COMCIDP     7150     A
COMCIDP  *                                                                               COMCIDP     7151     A
COMCIDP  *                SEARCHS A TABLE THAT IS TERMINATED BY A ZERO WORD              COMCIDP     7152     A
COMCIDP  *         FOR A MATCH, USING A SPECIFIED EXTRACTION MASK.                       COMCIDP     7153     A
COMCIDP  *                                                                               COMCIDP     7154     A
COMCIDP  *         ENTRY  (B6)     =  FWA TO BEGIN SEARCH                                COMCIDP     7155     A
COMCIDP  *                (X0)     =  EXTRACTION MASK TO USE                             COMCIDP     7156     A
COMCIDP  *                (X1)     =  ELEMENT TO SEARCH FOR                              COMCIDP     7157     A
COMCIDP  *                                                                               COMCIDP     7158     A
COMCIDP  *         EXIT   IF NO FIND --                                                  COMCIDP     7159     A
COMCIDP  *                (X2)     =  .ZR.                                               COMCIDP     7160     A
COMCIDP  *                IF A FIND --                                                  COMCIDP     7161     A
COMCIDP  *                (B7)     =  ORDINAL OF FIND                                    COMCIDP     7162     A
COMCIDP  *                (A2,X2)  =  ADDR + CONTENTS OF WORD THAT MATCHED               COMCIDP     7163     A
COMCIDP  *                                                                               COMCIDP     7164     A
COMCIDP  *         USES   X - 2,3,6,7                                                    COMCIDP     7165     A
COMCIDP  *                A - 2                                                          COMCIDP     7166     A
COMCIDP  *                B - 7                                                          COMCIDP     7167     A
COMCIDP  *                                                                               COMCIDP     7168     A
COMCIDP  *         CALLS  NONE                                                           COMCIDP     7169     A
COMCIDP                                                                                  COMCIDP     7170     A
COMCIDP                                                                                  COMCIDP     7171     A
COMCIDP   SKT      SUBR              ** ENTRY/EXIT **                                    COMCIDP     7172     A
COMCIDP            SA2    B6                                                             COMCIDP     7173     A
COMCIDP            BX6    X0*X1                                                          COMCIDP     7174     A
COMCIDP            SB7    -1                                                             COMCIDP     7175     A
COMCIDP                                                                                  COMCIDP     7176     A
COMCIDP   SKT2     ZR     X2,EXIT.    IF END OF TABLE                                    COMCIDP     7177     A
COMCIDP            BX7    X0*X2                                                          COMCIDP     7178     A
COMCIDP            SB7    B7+B1                                                          COMCIDP     7179     A
COMCIDP            IX3    X6-X7                                                          COMCIDP     7180     A
COMCIDP            ZR     X3,EXIT.    IF A FIND                                          COMCIDP     7181     A
COMCIDP            SA2    A2+B1                                                          COMCIDP     7182     A
COMCIDP            EQ     SKT2                                                           COMCIDP     7183     A
COMCIDP   SLF      SPACE  4,10                                                           COMCIDP     7184     A
COMCIDP   .IDPOS   IFNE   .OS,2                                                          CCGA018       35     A
COMCIDP  **        SLF - SEARCH FOR LOGICAL FILE NAME.                                   COMCIDP     7185     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *                                                                    COMCIDP    7186    A
COMCIDP  *                                                                    COMCIDP    7187    A
COMCIDP  *         *SLF* TRIES TO ASSOCIATE A FET FWA WITH A GIVEN DPC        COMCIDP    7188    A
COMCIDP  *         LFN BY --                                                  COMCIDP    7189    A
COMCIDP  *                                                                    COMCIDP    7190    A
COMCIDP  *           1. SEARCHING THE USER FILE/FET TABLE (UFT=) LIST IF IT   COMCIDP    7191    A
COMCIDP  *              WAS PROVIDED (=Y).                                    COMCIDP    7192    A
COMCIDP  *                                                                    COMCIDP    7193    A
COMCIDP  *           2. SEARCHING RA.SSW+2 THRU RA.SSW+20B TO SEE IF ANYTHING COMCIDP    7194    A
COMCIDP  *              HERE LOOKS LIKE A FILE/FET NAME.                      COMCIDP    7195    A
COMCIDP  *                                                                    COMCIDP    7196    A
COMCIDP  *         ENTRY  (X1) = LFN TO SEARCH FOR, -L- FMT.                  COMCIDP    7197    A
COMCIDP  *                                                                    COMCIDP    7198    A
COMCIDP  *         EXIT   (X6) = FWA OF FET FOR LFN, ELSE .ZR. IF NO FIND.    COMCIDP    7199    A
COMCIDP  *                                                                    COMCIDP    7200    A
COMCIDP  *         USES   ALL BUT A0,X0,A5,X5                                 COMCIDP    7201    A
COMCIDP  *                                                                    COMCIDP    7202    A
COMCIDP  *         CALLS  CHK                                                 COMCIDP    7203    A
COMCIDP                                                                       COMCIDP    7204    A
COMCIDP                                                                       COMCIDP    7205    A
COMCIDP   SLF      SUBR               ** ENTRY/EXIT **                        COMCIDP    7206    A
COMCIDP                                                                       COMCIDP    7207    A
COMCIDP  *         HERE TO TRY POSSIBLE FET VECTOR LIST PROVIDED BY HOST      COMCIDP    7208    A
COMCIDP  *         AT *UFT=* (USER FILE/FET TABLE).                           COMCIDP    7209    A
COMCIDP                                                                       COMCIDP    7210    A
COMCIDP            SB7    =YUFT=      (B7) = FWA OF USER FILE TABLE, IF AVAIL COMCIDP    7211    A
COMCIDP            LX7    X1                                                  COMCIDP    7212    A
COMCIDP            SA7    SLFA        SAVE LFN                                COMCIDP    7213    A
COMCIDP            MI     B7,SLF3     IF USER DID NOT PROVIDE *UFT=*          COMCIDP    7214    A
COMCIDP                                                                       COMCIDP    7215    A
COMCIDP   SLF2     SB2    B7          (B2) = ADDR OF NEXT *UFT=* ENTRY        COMCIDP    7216    A
COMCIDP            RJ     CHK         CHECK CM ADDR (*UFT=* ENTRY ADDR)       COMCIDP    7217    A
COMCIDP            MI     B2,SE.UFT   IF *UFT=* ENTRY OUT-OF-RANGE            COMCIDP    7218    A
COMCIDP                                                                       COMCIDP    7219    A
COMCIDP            SA1    B7          (X1) = NEXT *UFT=* ENTRY                COMCIDP    7220    A
COMCIDP            SB7    B7+B1                                               COMCIDP    7221    A
COMCIDP            SX6    X1          (X6) = FWA OF FET, ELSE .ZR.            COMCIDP    7222    A
COMCIDP            MX3    7*CHAR                                              COMCIDP    7223    A
COMCIDP            ZR     X1,SLF3     IF END-OF-TABLE, NO FIND...             COMCIDP    7224    A
COMCIDP            SA2    SLFA        (X2) = LFN TO SEARCH FOR, -L- FMT       COMCIDP    7225    A
COMCIDP            BX4    X3*X1       (X4) = LFN FROM *UFT=*, -L- FMT         COMCIDP    7226    A
COMCIDP            IX7    X2-X4                                               COMCIDP    7227    A
COMCIDP            ZR     X7,EXIT.    IF A FIND IN *UFT=*                     COMCIDP    7228    A
COMCIDP            SB2    X1          (B2) = FWA OF FET                       COMCIDP    7229    A
COMCIDP            RJ     CHK         CHECK CM ADDRESS (FET FWA)              COMCIDP    7230    A
COMCIDP            BX6    X6-X6                                               COMCIDP    7231    A
COMCIDP            MI     B2,SLF2     IF FET ADDR NO GOOD                     COMCIDP    7232    A
COMCIDP            SA1    B2          (X1) = WORD 1 OF FET                    COMCIDP    7233    A
COMCIDP            SA2    SLFA        (X2) = LFN TO SEARCH FOR, -L- FMT       COMCIDP    7234    A
COMCIDP            BX4    X3*X1       (X4) = FET LFN, -L- FMT                 COMCIDP    7235    A
COMCIDP            IX7    X2-X4                                               COMCIDP    7236    A
COMCIDP            SX6    B2+         (X6) = FWA OF FET                       COMCIDP    7237    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            ZR     X7,EXIT.    IF A FIND                                    COMCIDP     7238     A
COMCIDP            EQ     SLF2                                                     COMCIDP     7239     A
COMCIDP                                                                             COMCIDP     7240     A
COMCIDP  *         HERE TO TRY POSSIBLE FET VECTOR LIST AT RA+2 THRU RA+20.        COMCIDP     7241     A
COMCIDP                                                                             COMCIDP     7242     A
COMCIDP   SLF3     SB6    RA.SSW+2    (B6) = FWA OF POSSIBLE FET VECTOR            COMCIDP     7243     A
COMCIDP            SB7    RA.SSW+20B  (B7) = LWA OF POSSIBLE FET VECTOR            COMCIDP     7244     A
COMCIDP                                                                             COMCIDP     7245     A
COMCIDP   SLF4     SA1    B6          (X1) = NEXT POSSIBLE FET VECTOR ENTRY        COMCIDP     7246     A
COMCIDP            BX6    X6-X6                                                    COMCIDP     7247     A
COMCIDP            GT     B6,B7,EXIT. IF END OF ALL POSSIBLE VECTORS               COMCIDP     7248     A
COMCIDP            SB6    B6+B1                                                    COMCIDP     7249     A
COMCIDP            MX3    7*CHAR                                                   COMCIDP     7250     A
COMCIDP            SA2    SLFA        (X2) = LFN TO SEARCH FOR, -L- FMT            COMCIDP     7251     A
COMCIDP            BX4    X3*X1       (X4) = LFN FROM VECTOR, -L- FMT              COMCIDP     7252     A
COMCIDP            IX7    X2-X4                                                    COMCIDP     7253     A
COMCIDP            SX6    X1          (X6) = FWA OF FET                            COMCIDP     7254     A
COMCIDP            SB2    X1                                                       COMCIDP     7255     A
COMCIDP            ZR     X7,EXIT.    IF A FIND                                    COMCIDP     7256     A
COMCIDP            RJ     CHK         CHECK CM ADDRESS (FET FWA)                   COMCIDP     7257     A
COMCIDP            BX6    X6-X6                                                    COMCIDP     7258     A
COMCIDP            MI     B2,SLF4     IF FET ADDR NO GOOD                          COMCIDP     7259     A
COMCIDP            SA1    B2          (X1) = WORD 1 OF FET                         COMCIDP     7260     A
COMCIDP            SA2    SLFA        (X2) = LFN TO SEARCH FOR, -L- FMT            COMCIDP     7261     A
COMCIDP            BX4    X3*X1       (X4) = FET LFN, -L- FMT                      COMCIDP     7262     A
COMCIDP            IX7    X2-X4                                                    COMCIDP     7263     A
COMCIDP            SX6    B2+         (X6) = FWA OF FET                            COMCIDP     7264     A
COMCIDP            ZR     X7,EXIT.    IF A FIND                                    COMCIDP     7265     A
COMCIDP            EQ     SLF4                                                     COMCIDP     7266     A
COMCIDP                                                                             COMCIDP     7267     A
COMCIDP                                                                             COMCIDP     7268     A
COMCIDP   SLFA     BSSZ   1           SAVED LFN TO SEARCH FOR                      COMCIDP     7269     A
COMCIDP   .IDPOS   ENDIF                                                            CCGA018       36     A
COMCIDP   SOB      SPACE  4,10                                                     COMCIDP     7270     A
COMCIDP  **        SOB - SET OUTPUT BIT FLAGS.                                     COMCIDP     7271     A
COMCIDP  *                                                                          COMCIDP     7272     A
COMCIDP  *                                                                          COMCIDP     7273     A
COMCIDP  *                 THIS ROUTINE SETS BITS IN THE MASTER CONTROL FLAG,       COMCIDP     7274     A
COMCIDP  *         *IDPFLG*, WHICH DETERMINES WHERE OUTPUT IS TO BE WRITTEN BY A    COMCIDP     7275     A
COMCIDP  *         BATCH (I.E. NON-INTERACTIVE) EXECUTIVE --                        COMCIDP     7276     A
COMCIDP  *                                                                          COMCIDP     7277     A
COMCIDP  *           1. IF INTERACTIVE MODE HAS EVER BEEN ENTERED (BIT *FTO* ON),   COMCIDP     7278     A
COMCIDP  *                THEN NO FLAGS ARE SET  (I.E. ALL FLAGS ARE AS SET BY      COMCIDP     7279     A
COMCIDP  *                INTERACTIVE USER VIA *OUTPUT* COMMAND).                   COMCIDP     7280     A
COMCIDP  *           2. IF INTERACTIVE MODE HAS NOT YET BEEN ENTERED                COMCIDP     7281     A
COMCIDP  *                (BIT *FTO* OFF), THEN SET --                              COMCIDP     7282     A
COMCIDP  *                  BDO = WRITING TO BATCH DEBUG OUTPUT FILE, *F.BDO*       COMCIDP     7283     A
COMCIDP  *                  SNL = L.BSL                                             COMCIDP     7284     A
COMCIDP  *                                                                          COMCIDP     7285     A
COMCIDP  *         ENTRY  (X1) = BITS 0 THRU *IDF.XECL-1* CONTAIN THE VALUE TO      COMCIDP     7286     A
COMCIDP  *                       BE PLACED IN *XEC* FIELD,                          COMCIDP     7287     A
COMCIDP  *                       (BITS 59 THRU *IDF.XECL* ARE IGNORED).             COMCIDP     7288     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *                                                                  COMCIDP    7289    A
COMCIDP  *         EXIT   (IDPFLG) UPDATED                                  COMCIDP    7290    A
COMCIDP  *                                                                  COMCIDP    7291    A
COMCIDP  *         USES   X - 1,2,3,4,6                                     COMCIDP    7292    A
COMCIDP  *                A - 2,6                                           COMCIDP    7293    A
COMCIDP  *                B - NONE                                          COMCIDP    7294    A
COMCIDP  *                                                                  COMCIDP    7295    A
COMCIDP  *         CALLS  NONE                                              COMCIDP    7296    A
COMCIDP                                                                     COMCIDP    7297    A
COMCIDP                                                                     COMCIDP    7298    A
COMCIDP   SOB      SUBR                ** ENTRY/EXIT **                     COMCIDP    7299    A
COMCIDP            SA2    IDPFLG                                            COMCIDP    7300    A
COMCIDP            MX3    -IDF.XECL                                         COMCIDP    7301    A
COMCIDP            SX4    L.BSL                                             COMCIDP    7302    A
COMCIDP            BX1    -X3*X1                                            COMCIDP    7303    A
COMCIDP            LX2    0-IDF.XECP                                        COMCIDP    7304    A
COMCIDP            BX2    X3*X2       CLEAR OLD *XEC*                       COMCIDP    7305    A
COMCIDP            IX6    X2+X1       MERGE NEW *XEC*                       COMCIDP    7306    A
COMCIDP            LX2    59-IDF.FTOP+IDF.XECP-0                            COMCIDP    7307    A
COMCIDP            LX6    IDF.XECP-0                                        COMCIDP    7308    A
COMCIDP            MX3    1                                                 COMCIDP    7309    A
COMCIDP            SA6    A2                                                COMCIDP    7310    A
COMCIDP            MI     X2,EXIT.    IF *IDP* HAS EVER BEEN CALLED         COMCIDP    7311    A
COMCIDP            LX3    IDF.BDOP-59                                       COMCIDP    7312    A
COMCIDP            MX2    -IDF.SNLL                                         COMCIDP    7313    A
COMCIDP            BX6    X6+X3       MERGE *BDO*                           COMCIDP    7314    A
COMCIDP            LX2    IDF.SNLP-0                                        COMCIDP    7315    A
COMCIDP            LX4    IDF.SNLP-0                                        COMCIDP    7316    A
COMCIDP            BX6    X2*X6       CLEAR OLD *SNL*                       COMCIDP    7317    A
COMCIDP            IX6    X6+X4       MERGE NEW *SNL*                       COMCIDP    7318    A
COMCIDP            SA6    A2                                                COMCIDP    7319    A
COMCIDP            EQ     EXIT.                                             COMCIDP    7320    A
COMCIDP   SSY      SPACE  4,10                                              COMCIDP    7321    A
COMCIDP   .IDPOS   IFNE   .OS,2                                             CCGA018      37    A
COMCIDP  **        SSY - SEARCH SYMBOL TABLES.                              COMCIDP    7322    A
COMCIDP  *                                                                  COMCIDP    7323    A
COMCIDP  *                                                                  COMCIDP    7324    A
COMCIDP  *         *SSY* PERFORMS THE TASK OF TRYING TO ASSOCIATE A BINARY VALUE  COMCIDP  7325  A
COMCIDP  *         WITH A DPC NAME BY QUERYING IDP'S VARIOUS SYMBOL TABLES.  COMCIDP    7326    A
COMCIDP  *                                                                  COMCIDP    7327    A
COMCIDP  *         EACH SYMBOL HAS AN IMPLICIT SYMBOL TYPE ASSOCIATED WITH IT,  COMCIDP  7328  A
COMCIDP  *         PLACED INTO (B6) ON EXIT.  SYMBOL TYPES ARE DEFINED VIA  COMCIDP    7329    A
COMCIDP  *         COMPASS SYMBOLS OF THE FORM --                           COMCIDP    7330    A
COMCIDP  *                                                                  COMCIDP    7331    A
COMCIDP  *                SY.XXX                                            COMCIDP    7332    A
COMCIDP  *                                                                  COMCIDP    7333    A
COMCIDP  *         WHERE *SY.* IS A COMMON PREFIX, AND *XXX* IS A UNIQUE SYMBOL  COMCIDP  7334  A
COMCIDP  *         TYPE.  SYMBOL TYPES, *XXX* ARE --                        COMCIDP    7335    A
COMCIDP  *                                                                  COMCIDP    7336    A
COMCIDP  *           NONE = SYMBOL IS UNDEFINED.                            COMCIDP    7337    A
COMCIDP  *                                                                  COMCIDP    7338    A
COMCIDP  *           DECK = SYMBOL IS AN *FAA/RPV=RNA* DECK NAME.           COMCIDP    7339    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *                                                                  COMCIDP    7340    A
COMCIDP  *          SET  = SYMBOL WAS DEFINED VIA *SET* COMMAND.            COMCIDP    7341    A
COMCIDP  *                                                                  COMCIDP    7342    A
COMCIDP  *          LFN  = SYMBOL IS A LOGICAL FILE NAME.  I.E. IDP KNOWS   COMCIDP    7343    A
COMCIDP  *                 THE FWA OF ITS FET SOMEHOW.                      COMCIDP    7344    A
COMCIDP  *                                                                  COMCIDP    7345    A
COMCIDP  *          USY  = SYMBOL IS USER-DEFINED.                          COMCIDP    7346    A
COMCIDP  *                                                                  COMCIDP    7347    A
COMCIDP  *        ENTRY  (X1) = SYMBOL TO SEARCH FOR, -L- FMT.              COMCIDP    7348    A
COMCIDP  *                                                                  COMCIDP    7349    A
COMCIDP  *        EXIT   (B6) = SYMBOL TYPE (SEE ABOVE).  THAT IS, (B6) .NZ COMCIDP    7350    A
COMCIDP  *                      IF A FIND, ELSE .ZR.                        COMCIDP    7351    A
COMCIDP  *               (X6) = BINARY VALUE TO ASSOCIATE WITH THIS SYMBOL. COMCIDP    7352    A
COMCIDP  *                      NOTE THAT IF SYMBOL IS UNDEFINED, (B6) .ZR., COMCIDP  7353    A
COMCIDP  *                      THEN (X6) = 0.                              COMCIDP    7354    A
COMCIDP  *                                                                  COMCIDP    7355    A
COMCIDP  *        USES   ALL BUT A0,X0,A5,X5                                COMCIDP    7356    A
COMCIDP  *                                                                  COMCIDP    7357    A
COMCIDP  *        CALLS  FAA,SKT,USY=(IF AVAIL)                             COMCIDP    7358    A
COMCIDP                                                                     COMCIDP    7359    A
COMCIDP                                                                     COMCIDP    7360    A
COMCIDP   SSY     SUBR                ** ENTRY/EXIT **                      COMCIDP    7361    A
COMCIDP                                                                     COMCIDP    7362    A
COMCIDP  *        HERE TO TRY *SET* NAME.                                   COMCIDP    7363    A
COMCIDP                                                                     COMCIDP    7364    A
COMCIDP           BX6    X1                                                 COMCIDP    7365    A
COMCIDP           SB6    IDPSET      (B6) = FWA OF *SET* TABLE              COMCIDP    7366    A
COMCIDP           MX0    7*CHAR      (X0) = EXTRACTION MASK FOR *SKT*       COMCIDP    7367    A
COMCIDP           SA6    SSYA        SAVE (X1)                              COMCIDP    7368    A
COMCIDP           RJ     SKT         SEARCH *SET* TABLE FOR SYMBOL          COMCIDP    7369    A
COMCIDP           SB6    SY.SET      SET TO *SYMBOL IS SET NAME*            COMCIDP    7370    A
COMCIDP           SX6    X2+         (X6) = BINARY VALUE OF THIS SYMBOL     COMCIDP    7371    A
COMCIDP           NZ     X2,EXIT.    IF A FIND                              COMCIDP    7372    A
COMCIDP                                                                     COMCIDP    7373    A
COMCIDP  *        HERE TO TRY DECK NAME.                                    COMCIDP    7374    A
COMCIDP                                                                     COMCIDP    7375    A
COMCIDP           SA1    SSYA        (X1) = SYMBOL TO SEARCH FOR, -L- FMT   COMCIDP    7376    A
COMCIDP           MX0    7*CHAR      (X0) = EXTRACTION MASK FOR *FAA*       COMCIDP    7377    A
COMCIDP           RJ     FAA         FIND ABSOLUTE ADDRESS                  COMCIDP    7378    A
COMCIDP           SB6    SY.DECK     SET TO *SYMBOL IS DECK NAME*           COMCIDP    7379    A
COMCIDP           PL     X6,EXIT.    IF A FIND                              COMCIDP    7380    A
COMCIDP                                                                     COMCIDP    7381    A
COMCIDP  *        HERE TO TRY LFN/FET NAME.                                 COMCIDP    7382    A
COMCIDP                                                                     COMCIDP    7383    A
COMCIDP           SA1    SSYA        (X1) = SYMBOL TO SEARCH FOR, -L- FMT   COMCIDP    7384    A
COMCIDP           RJ     SLF         SEARCH FOR LFN                         COMCIDP    7385    A
COMCIDP           SB6    SY.LFN      SET TO *SYMBOL IS LFN*                 COMCIDP    7386    A
COMCIDP           NZ     X6,EXIT.    IF A FIND                              COMCIDP    7387    A
COMCIDP                                                                     COMCIDP    7388    A
COMCIDP  *        HERE TO TRY USER-DEFINED SYMBOL.                          COMCIDP    7389    A
COMCIDP                                                                     COMCIDP    7390    A
COMCIDP           SB2    =YUSY=                                             COMCIDP    7391    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            SA1    SSYA        (X1) = SYMBOL TO SEARCH FOR, -L- FMT                COMCIDP    7392    A
COMCIDP            =B6    SY.NONE     SET TO *NO FIND*                                    COMCIDP    7393    A
COMCIDP            BX6    X6-X6                                                           COMCIDP    7394    A
COMCIDP            MI     B2,EXIT.    IF USER DID NOT PROVIDE *USY=*                      COMCIDP    7395    A
COMCIDP            RJ     =YUSY=      USER SYMBOL SEARCH OWNCODE                          COMCIDP    7396    A
COMCIDP            SB6    SY.USY      SET TO *SYMBOL IS USER-DEFINED*                     COMCIDP    7397    A
COMCIDP            NZ     X1,EXIT.    IF A FIND                                           COMCIDP    7398    A
COMCIDP            =B6    SY.NONE     SET TO *SYMBOL IS UNDEFINED*                        COMCIDP    7399    A
COMCIDP            BX6    X6-X6                                                           COMCIDP    7400    A
COMCIDP            EQ     EXIT.                                                           COMCIDP    7401    A
COMCIDP                                                                                   COMCIDP    7402    A
COMCIDP                                                                                   COMCIDP    7403    A
COMCIDP    SSYA    BSSZ   1           SAVED SYMBOL TO SEARCH FOR                          COMCIDP    7404    A
COMCIDP    STP     SPACE  4,10                                                            COMCIDP    7405    A
COMCIDP   **       STP - STEP AN INSTRUCTION.                                             COMCIDP    7406    A
COMCIDP   *                                                                               COMCIDP    7407    A
COMCIDP   *                                                                               COMCIDP    7408    A
COMCIDP   *               THIS ROUTINE WILL STEP A SINGLE CPU INSTRUCTION, AND            COMCIDP    7409    A
COMCIDP   *        LIST THE RESULT REGISTER, IF APPROPRIATE.                              COMCIDP    7410    A
COMCIDP   *                                                                               COMCIDP    7411    A
COMCIDP   *               *STP* ALSO PERFORMS A LIMITED SET OF ERROR CHECKS FOR           COMCIDP    7412    A
COMCIDP   *        RANGE ERRORS (MODE=1). 1) THE PSEUDO P REGISTER IS CHECKED,            COMCIDP    7413    A
COMCIDP   *        2) XFERS OF CONTROL (BRANCH OR RJ) ARE CHECKED, AND                    COMCIDP    7414    A
COMCIDP   *        3) LOAD/STORE ADDRESSES ARE CHECKED. IF AN ERROR IS DETECTED,          COMCIDP    7415    A
COMCIDP   *        THE BAD INSTRUCTION AND THE -P- REGISTER ARE DISPLAYED (EVEN           COMCIDP    7416    A
COMCIDP   *        IF -L SELECTED). STEP MODE IS THEN TERMINATED.                         COMCIDP    7417    A
COMCIDP   *                                                                               COMCIDP    7418    A
COMCIDP   *        ENTRY  (BC=BRAD) = CONTAINS THE PREVIOUS CONTENTS OF IDP               COMCIDP    7419    A
COMCIDP   *                           GENERATED BREAK ADDR IF THEY NEED TO                COMCIDP    7420    A
COMCIDP   *                           BE XEQ, ELSE .ZR.                                   COMCIDP    7421    A
COMCIDP   *               (IDPPREG) = PSEUDO P REGISTER, I.E. ADDRESS OF WORD             COMCIDP    7422    A
COMCIDP   *                           CONTAINING INSTRUCTION TO XEQ                       COMCIDP    7423    A
COMCIDP   *               (IDPPOS)  = POSITION COUNTER. LEFT MOST BIT OF INSTR            COMCIDP    7424    A
COMCIDP   *                           TO XEQ IS BIT (IDPPOS)-1 IN ((IDPPREG)).            COMCIDP    7425    A
COMCIDP   *                           (BITS ARE NUMBERED 59 THRU 0)                       COMCIDP    7426    A
COMCIDP   *                           E.G. FOR A WORD CONTAINING--                        COMCIDP    7427    A
COMCIDP   *                            +  MX0  -CHAR                                      COMCIDP    7428    A
COMCIDP   *                               SA1  ADDR                                       COMCIDP    7429    A
COMCIDP   *                               BX6  -X0*X5                                     COMCIDP    7430    A
COMCIDP   *                           WHEN STEPPING *SA1  ADDR*--                         COMCIDP    7431    A
COMCIDP   *                           (IDPPOS) = 45                                       COMCIDP    7432    A
COMCIDP   *               (IDPXLST) = .NZ. IF LISTING EACH STEPPED INSTRUCTION,           COMCIDP    7433    A
COMCIDP   *                               ELSE .ZR.                                       COMCIDP    7434    A
COMCIDP   *                                                                               COMCIDP    7435    A
COMCIDP   *        EXIT   (BC=BRAD), (IDPPREG), AND (IDPPOS) UPDATED                      COMCIDP    7436    A
COMCIDP   *               TO *IDP=ER* IF AN ERROR DETECTED                                COMCIDP    7437    A
COMCIDP   *               TO *IDP2A* IF AN *RJ IDP=* IS ENCOUNTERED AND THE               COMCIDP    7438    A
COMCIDP   *                 BREAK CONDITIONS ARE STATISFIED                               COMCIDP    7439    A
COMCIDP   *                                                                               COMCIDP    7440    A
COMCIDP   *        USES   ALL BUT A0                                                      COMCIDP    7441    A
COMCIDP   *                                                                               COMCIDP    7442    A
COMCIDP   *        CALLS  BRK,CHK,DAB,DAZ,DUX,FRA=(IF DEF),FRK,                           COMCIDP    7443    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *                PRIDP,RIR,SVR                                          COMCIDP    7444    A
COMCIDP                                                                          COMCIDP    7445    A
COMCIDP                                                                          COMCIDP    7446    A
COMCIDP   STP      SUBR                ** ENTRY/EXIT **                          COMCIDP    7447    A
COMCIDP                                                                          COMCIDP    7448    A
COMCIDP  *         INITIALIZATION.                                               COMCIDP    7449    A
COMCIDP                                                                          COMCIDP    7450    A
COMCIDP            SX6    -1                                                     COMCIDP    7451    A
COMCIDP            SA6    IDPXFR      CLEAR *TRANSFER CONTROL* FLAG              COMCIDP    7452    A
COMCIDP            SA6    IDPXAR      SET TO *NO ADDR REF SO FAR*                COMCIDP    7453    A
COMCIDP                                                                          COMCIDP    7454    A
COMCIDP  *         FETCH WORD CONTAINING INSTRUCTION TO BE XEQ.                  COMCIDP    7455    A
COMCIDP                                                                          COMCIDP    7456    A
COMCIDP            SA1    IDPPREG     (X1) = PSEUDO P REGISTER                   COMCIDP    7457    A
COMCIDP            SA2    BC=BRAD                                                COMCIDP    7458    A
COMCIDP            SA3    A1+B1       (X3) = POS COUNTER                         COMCIDP    7459    A
COMCIDP            SA5    X1          (X5) = WORD CONTAINING INSTRUCTION TO XEQ  COMCIDP    7460    A
COMCIDP            ZR     X2,STP2     IF NOT *IDP* GENERATED BREAK               COMCIDP    7461    A
COMCIDP                                                                          COMCIDP    7462    A
COMCIDP  *         HERE BECAUSE INSTRUCTION TO XEQ AT *BC=BRAD*.                 COMCIDP    7463    A
COMCIDP                                                                          COMCIDP    7464    A
COMCIDP            SA5    A5-B1       (A5) = ADDR OF GENERATED BREAK             COMCIDP    7465    A
COMCIDP            BX5    X2                                                     COMCIDP    7466    A
COMCIDP                                                                          COMCIDP    7467    A
COMCIDP  *         LIST INSTRUCTION IF NECESSARY.                                COMCIDP    7468    A
COMCIDP                                                                          COMCIDP    7469    A
COMCIDP   STP2     SA1    IDPXLST                                                COMCIDP    7470    A
COMCIDP            SB4    X3                                                     COMCIDP    7471    A
COMCIDP            ZR     X1,STP3     IF NOT LISTING INSTRUCTION                 COMCIDP    7472    A
COMCIDP            RJ     DAZ         DEASSEMBLE INSTRUCTION                     COMCIDP    7473    A
COMCIDP                                                                          COMCIDP    7474    A
COMCIDP  *         BREAK OUT OPCODE.                                             COMCIDP    7475    A
COMCIDP                                                                          COMCIDP    7476    A
COMCIDP   STP3     SB6    -B4                                                    COMCIDP    7477    A
COMCIDP            MX0    -6                                                     COMCIDP    7478    A
COMCIDP            SB6    B6+60+6                                                COMCIDP    7479    A
COMCIDP            LX5    X5,B6       (X5) = 54/OTHER STUFF, 6/OPCODE            COMCIDP    7480    A
COMCIDP            BX6    -X0*X5                                                 COMCIDP    7481    A
COMCIDP            LX5    3           (X5) = 51/OTHER, 6/OPC, 3/I                COMCIDP    7482    A
COMCIDP            MX0    -3                                                     COMCIDP    7483    A
COMCIDP            SA6    STP=OP                                                 COMCIDP    7484    A
COMCIDP            ZR     X6,STP=ER   IF 00B OPCODE, TRYING TO XEQ DATA...       COMCIDP    7485    A
COMCIDP                                                                          COMCIDP    7486    A
COMCIDP  *         BREAK OUT *I* FIELD.                                          COMCIDP    7487    A
COMCIDP                                                                          COMCIDP    7488    A
COMCIDP            BX6    -X0*X5                                                 COMCIDP    7489    A
COMCIDP            LX5    3           (X5) = 48/OTHER, 6/OPC, 3/I, 3/J           COMCIDP    7490    A
COMCIDP            SA6    A6+B1                                                  COMCIDP    7491    A
COMCIDP                                                                          COMCIDP    7492    A
COMCIDP  *         BREAK OUT *J* FIELD.                                          COMCIDP    7493    A
COMCIDP                                                                          COMCIDP    7494    A
COMCIDP            BX6    -X0*X5                                                 COMCIDP    7495    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            LX5    3           (X5) = 45/OTHER, 6/OPC, 3/I, 3/J, 3/K          COMCIDP    7496    A
COMCIDP            SA6    A6+B1                                                      COMCIDP    7497    A
COMCIDP                                                                              COMCIDP    7498    A
COMCIDP  *         BREAK OUT *K* FIELD.                                               COMCIDP    7499    A
COMCIDP                                                                              COMCIDP    7500    A
COMCIDP            BX6    -X0*X5                                                     COMCIDP    7501    A
COMCIDP            LX5    18-3        (X5) = 30/OTHER, 6/OPC, 3/I, 3/J, 18/Q         COMCIDP    7502    A
COMCIDP            SA6    A6+B1                                                      COMCIDP    7503    A
COMCIDP                                                                              COMCIDP    7504    A
COMCIDP  *         BREAK OUT *Q* 18 BIT ADDRESS FIELD.                                COMCIDP    7505    A
COMCIDP                                                                              COMCIDP    7506    A
COMCIDP            SX6    X5                                                         COMCIDP    7507    A
COMCIDP            SA6    STP=ADR                                                    COMCIDP    7508    A
COMCIDP                                                                              COMCIDP    7509    A
COMCIDP  *         FETCH INSTRUCTION SKELETON AND CHECK FOR LENGTH CONFLICT.          COMCIDP    7510    A
COMCIDP                                                                              COMCIDP    7511    A
COMCIDP            SA1    STP=OP                                                     COMCIDP    7512    A
COMCIDP            LX5    30          LEFT JUSTIFY INSTRUCTION                       COMCIDP    7513    A
COMCIDP            MX0    15                                                         COMCIDP    7514    A
COMCIDP            SA2    DAZ=PS+X1   (X2) = INSTRUCTION SKELETON                    COMCIDP    7515    A
COMCIDP            SB5    15                                                         COMCIDP    7516    A
COMCIDP            PL     X2,STP4     IF SHORT (15 BIT) INSTRUCTION                  COMCIDP    7517    A
COMCIDP            MX0    30                                                         COMCIDP    7518    A
COMCIDP            SB5    B5+B5                                                      COMCIDP    7519    A
COMCIDP                                                                              COMCIDP    7520    A
COMCIDP   STP4     LT     B4,B5,STP=ER IF NOT ENOUGH ROOM FOR 30 BIT INSTRUCTION,    COMCIDP    7521    A
COMCIDP  *                             TRYING TO XEQ DATA...                         COMCIDP    7522    A
COMCIDP                                                                              COMCIDP    7523    A
COMCIDP  *         BREAK OUT FULL INSTRUCTION.                                        COMCIDP    7524    A
COMCIDP                                                                              COMCIDP    7525    A
COMCIDP            SA3    =4600046000460004600 0B                                    COMCIDP    7526    A
COMCIDP            BX6    X0*X5                                                      COMCIDP    7527    A
COMCIDP            BX7    -X0*X3                                                     COMCIDP    7528    A
COMCIDP            IX6    X6+X7       (X6) = INSTRUCTION, LEFT JUST W/ NO-OP FILL    COMCIDP    7529    A
COMCIDP            SA6    STP=IN                                                     COMCIDP    7530    A
COMCIDP                                                                              COMCIDP    7531    A
COMCIDP  *         SPLIT INSTRUCTION INTO REPRESENTATIVE TYPE--                       COMCIDP    7532    A
COMCIDP  *           01 THRU 07  BRANCH                                               COMCIDP    7533    A
COMCIDP  *           10 THRU 47  NORMAL                                               COMCIDP    7534    A
COMCIDP  *           50 THRU 57  LOAD/STORE                                           COMCIDP    7535    A
COMCIDP  *           60 THRU 77  NORMAL                                               COMCIDP    7536    A
COMCIDP                                                                              COMCIDP    7537    A
COMCIDP            SX3    X1-10B                                                     COMCIDP    7538    A
COMCIDP            SX4    X1-50B                                                     COMCIDP    7539    A
COMCIDP            MI     X3,STP6     IF A BRANCH                                    COMCIDP    7540    A
COMCIDP            MI     X4,STP5     IF A NORMAL                                    COMCIDP    7541    A
COMCIDP            SX3    X1-60B                                                     COMCIDP    7542    A
COMCIDP            MI     X3,STP12    IF A LOAD/STORE                                COMCIDP    7543    A
COMCIDP                                                                              COMCIDP    7544    A
COMCIDP  *         HERE IF A NORMAL INSTRUCTION (NO SPECIAL PROCESSING).              COMCIDP    7545    A
COMCIDP                                                                              COMCIDP    7546    A
COMCIDP   STP5     SX1    FW.SVR      (X1) = FWA OF SAVED REGISTERS                  COMCIDP    7547    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            RJ     RIR         RESTORE INTERACTIVE REGISTERS                    COMCIDP     7548     A
COMCIDP                                                                                COMCIDP     7549     A
COMCIDP  *         HERE TO XEQ NORMAL INSTRUCTION.                                     COMCIDP     7550     A
COMCIDP                                                                                COMCIDP     7551     A
COMCIDP   STP=IN   BSSZ   1            THIS WORD IS PLUGGED WITH THE INSTRUCTION       COMCIDP     7552     A
COMCIDP  *                               TO BE XEQ, LEFT JUSTIFIED WITH NO-OP FILL     COMCIDP     7553     A
COMCIDP                                                                                COMCIDP     7554     A
COMCIDP   +        RJ     =XSVR       SAVE ORIGINAL REGISTERS                          COMCIDP     7555     A
COMCIDP   -        VFD    30/FW.SVR                                                    COMCIDP     7556     A
COMCIDP            EQ     STP=X                                                        COMCIDP     7557     A
COMCIDP                                                                                COMCIDP     7558     A
COMCIDP  *         HERE IF A BRANCH INSTRUCTION (01 THRU 07).                          COMCIDP     7559     A
COMCIDP                                                                                COMCIDP     7560     A
COMCIDP   STP6     SA2    STP=I                                                        COMCIDP     7561     A
COMCIDP            SX6    X1-01B                                                       COMCIDP     7562     A
COMCIDP            SA3    STP=ADR                                                      COMCIDP     7563     A
COMCIDP            NZ     X6,STP8     IF NOT 01B INSTRUCTION                           COMCIDP     7564     A
COMCIDP            NZ     X2,STP5     IF NOT *RJ*                                      COMCIDP     7565     A
COMCIDP                                                                                COMCIDP     7566     A
COMCIDP  *         HERE IF AN *RJ* INSTRUCTION.                                        COMCIDP     7567     A
COMCIDP                                                                                COMCIDP     7568     A
COMCIDP            SA4    IDPPREG                                                      COMCIDP     7569     A
COMCIDP            SA5    BC=BRAD     (X5) = WORD CONTAINING *RJ* INSTRUCTION          COMCIDP     7570     A
COMCIDP  *                                   (IF PREVIOUS CONTENTS OF GEN BRK)         COMCIDP     7571     A
COMCIDP            BX6    X4          (X6) = /*+1/                                     COMCIDP     7572     A
COMCIDP            SX7    X3-REG=                                                      COMCIDP     7573     A
COMCIDP            SX0    X3-SNP=                                                      COMCIDP     7574     A
COMCIDP            NZ     X5,STP6A    IF *RJ* IS PREVIOUS CONTENTS OF BREAK ADDR       COMCIDP     7575     A
COMCIDP            SA5    X4          (X5) = WORD CONTAINING *RJ* INSTRUCTION          COMCIDP     7576     A
COMCIDP            SX6    X4+B1       (X6) = /*+1/                                     COMCIDP     7577     A
COMCIDP                                                                                COMCIDP     7578     A
COMCIDP  *         CHECK FOR CALL TO *REG=/SNP=*.  THESE ROUTINES MUST AVOID           COMCIDP     7579     A
COMCIDP  *           BEING STEPPED BECAUSE THEY USE *SVR=/RSR=* TO SAVE AND            COMCIDP     7580     A
COMCIDP  *           RESTORE THE CALLERS REGISTERS (SEE NEXT PARAGRAPH).               COMCIDP     7581     A
COMCIDP                                                                                COMCIDP     7582     A
COMCIDP   STP6A    ZR     X7,STP7     IF *RJ REG=*                                     COMCIDP     7583     A
COMCIDP            ZR     X0,STP7     IF *RJ SNP=*                                     COMCIDP     7584     A
COMCIDP                                                                                COMCIDP     7585     A
COMCIDP  *         CHECK FOR CALL TO *RIR/RSR/SVR/SYS=/XJR*.                           COMCIDP     7586     A
COMCIDP  *                                                                             COMCIDP     7587     A
COMCIDP  *         THIS IS CURRENTLY FATALLY RECURSIVE BECAUSE *STP* USES THESE        COMCIDP     7588     A
COMCIDP  *         ROUTINES TO SAVE AND RESTORE THE IDP CALLER'S REGISTERS             COMCIDP     7589     A
COMCIDP  *         BEFORE AND AFTER EACH INSTRUCTION IS STEPPED.  THIS IMPLIES         COMCIDP     7590     A
COMCIDP  *         THAT THESE *SAVE AND RESTORE* ROUTINES WOULD HAVE TO BE             COMCIDP     7591     A
COMCIDP  *         ABLE TO BE STEPPED AND EXECUTED CONCURRENTLY...TRICKY.              COMCIDP     7592     A
COMCIDP  *                                                                             COMCIDP     7593     A
COMCIDP  *         IN CASE YOU ARE CONFUSED ABOUT *SYS=*, NOTE THAT *COMCXJR*          COMCIDP     7594     A
COMCIDP  *         CALLS *SYS=* TO ISSUE *XJR* RA+1 REQUEST.                           COMCIDP     7595     A
COMCIDP                                                                                COMCIDP     7596     A
COMCIDP            SX1    =XSVR                                                        COMCIDP     7597     A
COMCIDP            SX2    RIR                                                          COMCIDP     7598     A
COMCIDP            IX1    X1-X3                                                        COMCIDP     7599     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            IX2    X2-X3                                                  COMCIDP    7600    A
COMCIDP            ZR     X1,STP=ER   IF *RJ SVR*, FATAL RECURSION...            COMCIDP    7601    A
COMCIDP            ZR     X2,STP=ER   IF *RJ RIR*, FATAL RECURSION...            COMCIDP    7602    A
COMCIDP            SX1    =XRSR                                                  COMCIDP    7603    A
COMCIDP            SX2    =XXJR                                                  COMCIDP    7604    A
COMCIDP            IX1    X1-X3                                                  COMCIDP    7605    A
COMCIDP            IX2    X2-X3                                                  COMCIDP    7606    A
COMCIDP            ZR     X1,STP=ER   IF *RJ RSR*, FATAL RECURSION...            COMCIDP    7607    A
COMCIDP            ZR     X2,STP=ER   IF *RJ XJR*, FATAL RECURSION...            COMCIDP    7608    A
COMCIDP            SX1    =XSYS=                                                 COMCIDP    7609    A
COMCIDP            IX1    X1-X3                                                  COMCIDP    7610    A
COMCIDP            ZR     X1,STP=ER   IF *RJ SYS=*, FATAL RECURSION...           COMCIDP    7611    A
COMCIDP                                                                           COMCIDP    7612    A
COMCIDP  *         SET UP TRANSFER ADDRESS OF /RJ Q/ TO BE /Q+1/.                 COMCIDP    7613    A
COMCIDP                                                                           COMCIDP    7614    A
COMCIDP            SX7    X3+B1       (X7) = /Q+1/                               COMCIDP    7615    A
COMCIDP            SB2    X7                                                     COMCIDP    7616    A
COMCIDP            SA7    IDPXFR                                                 COMCIDP    7617    A
COMCIDP                                                                           COMCIDP    7618    A
COMCIDP  *         CHECK TRANSFER ADDR AND SIMULATE *RJ* BY PLUGGING /EQ *+1/.    COMCIDP    7619    A
COMCIDP  *                                                                        COMCIDP    7620    A
COMCIDP  *         ENTRY  (B2) = TRANSFER ADDRESS                                 COMCIDP    7621    A
COMCIDP  *                (X6) = /*+1/  (I.E. RETURN ADDR)                        COMCIDP    7622    A
COMCIDP  *                (X3) = /Q/    (I.E. ADDR TO PLUG /EQ *+1/)              COMCIDP    7623    A
COMCIDP  *                                                                        COMCIDP    7624    A
COMCIDP  *         NOTE   WHEN THIS IS NOT *RJ REG=/SNP=*, THEN                   COMCIDP    7625    A
COMCIDP  *                TRANSFER ADDR = Q+1  OR  (B2) = (X3)+1.                 COMCIDP    7626    A
COMCIDP  *                                                                        COMCIDP    7627    A
COMCIDP  *                WHEN THIS IS AN *RJ REG=/SNP=*, THEN                   COMCIDP    7628    A
COMCIDP  *                TRANSFER ADDR = *+1  OR  (B2) = (X6), I.E. THE *RJ* HAS COMCIDP    7629    A
COMCIDP  *                BEEN SKIPPED.                                          COMCIDP    7630    A
COMCIDP                                                                           COMCIDP    7631    A
COMCIDP   STP6B    RJ     CHK         CHECK CM ADDRESS                           COMCIDP    7632    A
COMCIDP            SX7    0400B       (X7) = *EQ* INSTRUCTION                    COMCIDP    7633    A
COMCIDP            MI     B2,STP=ER   IF ADDR IS BAD                             COMCIDP    7634    A
COMCIDP            LX7    29-11                                                  COMCIDP    7635    A
COMCIDP            SX0    IDP=                                                   COMCIDP    7636    A
COMCIDP            BX7    X7+X6       (X7) = 30/0,30/EQ *+1                      COMCIDP    7637    A
COMCIDP            LX7    59-29       (X7) = 30/EQ *+1,30/0                      COMCIDP    7638    A
COMCIDP            SA7    X3                                                     COMCIDP    7639    A
COMCIDP                                                                           COMCIDP    7640    A
COMCIDP  *         CHECK FOR *RJ IDP=*.                                           COMCIDP    7641    A
COMCIDP                                                                           COMCIDP    7642    A
COMCIDP            IX0    X3-X0                                                  COMCIDP    7643    A
COMCIDP            SA1    IDPXRJ                                                 COMCIDP    7644    A
COMCIDP            ZR     X0,STP7A    IF *RJ IDP=*                               COMCIDP    7645    A
COMCIDP                                                                           COMCIDP    7646    A
COMCIDP  *         SET UP LEVEL 0 *RJ* ADDRESS.                                   COMCIDP    7647    A
COMCIDP                                                                           COMCIDP    7648    A
COMCIDP            NZ     X1,STP=X    IF NOT AT LEVEL 0 *RJ*                     COMCIDP    7649    A
COMCIDP            SX6    X3                                                     COMCIDP    7650    A
COMCIDP            SA6    A1                                                     COMCIDP    7651    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            EQ     STP=X                                                  COMCIDP     7652     A
COMCIDP                                                                           COMCIDP     7653     A
COMCIDP  *         HERE TO PROCESS *RJ REG=/SNP=*.                                COMCIDP     7654     A
COMCIDP                                                                           COMCIDP     7655     A
COMCIDP   STP7     SA6    IDPXFR      SET TO *RJ REG=/SNP= BECOMES                COMCIDP     7656     A
COMCIDP  *                              BRANCH TO /*+1/*                          COMCIDP     7657     A
COMCIDP            BX7    X5          (X7) = 30/RJ REG=/SNP=, 30/FWA OF PARM LIST COMCIDP     7658     A
COMCIDP            SA7    STP=SNP                                                 COMCIDP     7659     A
COMCIDP            SX1    FW.SVR      (X1) = FWA OF SAVED REGISTERS               COMCIDP     7660     A
COMCIDP            RJ     RIR         RESTORE INTERACTIVE REGISTERS               COMCIDP     7661     A
COMCIDP                                                                           COMCIDP     7662     A
COMCIDP   STP=SNP  BSSZ   1           THIS WORD IS PLUGGED WITH AN *RJ REG=* OR   COMCIDP     7663     A
COMCIDP  *                              AN *RJ SNP=* FORCED UPPER WITH THE FWA OF COMCIDP     7664     A
COMCIDP  *                              THE PARAMETER LIST IN THE LOWER 18 BITS   COMCIDP     7665     A
COMCIDP                                                                           COMCIDP     7666     A
COMCIDP            SA1    IDPXFR      (X1) = ADDR TO XFER TO (I.E. /*+1/)         COMCIDP     7667     A
COMCIDP            SA3    STP=ADR     (X3) = *REG=* OR *SNP=*                     COMCIDP     7668     A
COMCIDP            SB2    X1                                                      COMCIDP     7669     A
COMCIDP            BX6    X1                                                      COMCIDP     7670     A
COMCIDP            EQ     STP6B                                                   COMCIDP     7671     A
COMCIDP                                                                           COMCIDP     7672     A
COMCIDP  *         HERE TO PROCESS *RJ IDP=*.                                     COMCIDP     7673     A
COMCIDP                                                                           COMCIDP     7674     A
COMCIDP   STP7A    SA6    IDPXFR      SET TO */RJ IDP=/ BECOMES BRANCH TO /*+1/*  COMCIDP     7675     A
COMCIDP            SA1    IDP=        (A1,X1) = A+C OF IDP ENTRY POINT            COMCIDP     7676     A
COMCIDP            RJ     BRK         BREAK PROCESSOR                             COMCIDP     7677     A
COMCIDP            ZR     X5,STP=X    IF NO BREAK THIS TIME                       COMCIDP     7678     A
COMCIDP            RJ     FRK         CHECK FREQUENCY PARAMETERS                  COMCIDP     7679     A
COMCIDP            ZR     X5,STP=X    IF NO BREAK THIS TIME                       COMCIDP     7680     A
COMCIDP            EQ     IDP2A       HONOR THIS BREAK...                         COMCIDP     7681     A
COMCIDP                                                                           COMCIDP     7682     A
COMCIDP  *         HERE IF BRANCH INSTRUCTION (02 THRU 07).                       COMCIDP     7683     A
COMCIDP                                                                           COMCIDP     7684     A
COMCIDP   STP8     SX6    X1-4                                                    COMCIDP     7685     A
COMCIDP            SX7    X1-2                                                    COMCIDP     7686     A
COMCIDP            ZR     X6,STP10    IF *EQ BI,BJ,ADDR*                          COMCIDP     7687     A
COMCIDP            ZR     X7,STP11    IF *JP BI+ADDR*                             COMCIDP     7688     A
COMCIDP                                                                           COMCIDP     7689     A
COMCIDP  *         HERE IF CONDITIONAL BRANCH INSTRUCTION.                        COMCIDP     7690     A
COMCIDP                                                                           COMCIDP     7691     A
COMCIDP   STP9     SA1    STP=IN                                                  COMCIDP     7692     A
COMCIDP            MX0    -18                                                     COMCIDP     7693     A
COMCIDP            LX0    48-18                                                   COMCIDP     7694     A
COMCIDP            SX2    STP=BP      (X2) = BRANCH *PASS* ADDR                   COMCIDP     7695     A
COMCIDP            LX2    48-18                                                   COMCIDP     7696     A
COMCIDP            BX1    X0*X1                                                   COMCIDP     7697     A
COMCIDP            BX2    -X0*X2                                                  COMCIDP     7698     A
COMCIDP            IX6    X1+X2       (X6) = 12/BRANCH OPC, 18/STP=BRP, 30/NO-OPS COMCIDP     7699     A
COMCIDP            SA6    STP=BR                                                  COMCIDP     7700     A
COMCIDP                                                                           COMCIDP     7701     A
COMCIDP  *         HERE TO XEQ CONDITIONAL BRANCH INSTRUCTION. THE 18 BIT ADDRESS COMCIDP     7702     A
COMCIDP  *           *Q* PORTION OF THE BRANCH INSTRUCTION IS MODIFIED SO THAT IF COMCIDP     7703     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *          THE BRANCH IS TAKEN, CONTROL PASSES TO *STP=BP*.                COMCIDP   7704   A
COMCIDP                                                                             COMCIDP   7705   A
COMCIDP            SX1    FW.SVR      (X1) = FWA OF SAVED REGISTERS                 COMCIDP   7706   A
COMCIDP            RJ     RIR         RESTORE INTERACTIVE REGISTERS                 COMCIDP   7707   A
COMCIDP                                                                             COMCIDP   7708   A
COMCIDP   STP=BR   BSSZ   1           BRANCH TO XEQ IS PLUGGED HERE -- LEFT JUST-   COMCIDP   7709   A
COMCIDP  *                              IFIED W/ *Q* ALTERED AND NO-OP FILL         COMCIDP   7710   A
COMCIDP                                                                             COMCIDP   7711   A
COMCIDP   STP=BF   SB1    1           RESTORE (B1) = 1                              COMCIDP   7712   A
COMCIDP            EQ     STP=X                                                     COMCIDP   7713   A
COMCIDP                                                                             COMCIDP   7714   A
COMCIDP   STP=BP   SA1    STP=ADR                                                   COMCIDP   7715   A
COMCIDP            SB1    1           RESTORE (B1) = 1                              COMCIDP   7716   A
COMCIDP            SB2    X1          (B2) = ADDR TO TRANSFER CONTROL TO            COMCIDP   7717   A
COMCIDP            SX6    X1                                                        COMCIDP   7718   A
COMCIDP            SA6    IDPXFR                                                    COMCIDP   7719   A
COMCIDP            RJ     CHK         CHECK CM ADDRESS                              COMCIDP   7720   A
COMCIDP            SA1    IDPXLST                                                   COMCIDP   7721   A
COMCIDP            MI     B2,STP=ER   IF ADDR IS BAD                                COMCIDP   7722   A
COMCIDP            ZR     X1,STP=X    IF NOT LISTING                                COMCIDP   7723   A
COMCIDP            SA1    STP=I                                                     COMCIDP   7724   A
COMCIDP            SA2    A1+B1                                                     COMCIDP   7725   A
COMCIDP            BX6    X1+X2                                                     COMCIDP   7726   A
COMCIDP            ZR     X6,STP=X    IF *EQ ADDR*, DONT BE VERBOSE...              COMCIDP   7727   A
COMCIDP            PRIDP  (=C=                                BRANCH TAKEN...=)     COMCIDP   7728   A
COMCIDP            EQ     STP=X                                                     COMCIDP   7729   A
COMCIDP                                                                             COMCIDP   7730   A
COMCIDP  *         HERE IF *EQ BI,BJ,ADDR*.                                         COMCIDP   7731   A
COMCIDP                                                                             COMCIDP   7732   A
COMCIDP   STP10    SA1    STP=I                                                     COMCIDP   7733   A
COMCIDP            SA2    STP=J                                                     COMCIDP   7734   A
COMCIDP            BX6    X1+X2                                                     COMCIDP   7735   A
COMCIDP            NZ     X6,STP9     IF NOT *EQ ADDR*                              COMCIDP   7736   A
COMCIDP            MX7    0                                                         COMCIDP   7737   A
COMCIDP            SA1    IDPXRJ                                                    COMCIDP   7738   A
COMCIDP            SA2    IDPPREG                                                   COMCIDP   7739   A
COMCIDP            IX6    X1-X2                                                     COMCIDP   7740   A
COMCIDP            NZ     X6,STP=BP   IF NOT POSSIBLY *RJ SUBR* EXIT                COMCIDP   7741   A
COMCIDP            SA7    A1          SET TO *BACK TO LEVEL 0 NOW*                  COMCIDP   7742   A
COMCIDP            EQ     STP=BP                                                    COMCIDP   7743   A
COMCIDP                                                                             COMCIDP   7744   A
COMCIDP  *         HERE IF *JP BI+ADDR*.                                            COMCIDP   7745   A
COMCIDP                                                                             COMCIDP   7746   A
COMCIDP   STP11    SA1    STP=I                                                     COMCIDP   7747   A
COMCIDP            SA2    STP=ADR                                                   COMCIDP   7748   A
COMCIDP            SA3    IDP=SVB+X1   (X3) = (BI)                                  COMCIDP   7749   A
COMCIDP            IX6    X2+X3       (X6) = BI+ADDR                                COMCIDP   7750   A
COMCIDP            SB2    X6                                                        COMCIDP   7751   A
COMCIDP            SA6    IDPXFR                                                    COMCIDP   7752   A
COMCIDP            RJ     CHK         CHECK CM ADDRESS                              COMCIDP   7753   A
COMCIDP            PL     B2,STP=X    IF ADDR OK                                    COMCIDP   7754   A
COMCIDP            EQ     STP=ER                                                    COMCIDP   7755   A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP                                                                                  COMCIDP     7756     A
COMCIDP  *         HERE IF A LOAD/STORE INSTRUCTION (50 THRU 57).                        COMCIDP     7757     A
COMCIDP                                                                                  COMCIDP     7758     A
COMCIDP   STP12    SA1    STP=LS+X1-50B  (X1) = LOAD/STORE SKELETON                      COMCIDP     7759     A
COMCIDP            SA2    STP=I                                                          COMCIDP     7760     A
COMCIDP            SA3    A2+B1                                                          COMCIDP     7761     A
COMCIDP            SA4    A3+B1                                                          COMCIDP     7762     A
COMCIDP            SA5    STP=ADR                                                        COMCIDP     7763     A
COMCIDP            SB6    X1          (B6) = .MI. IF *Q* FLAG, ELSE                      COMCIDP     7764     A
COMCIDP  *                                = FWA OF APPROPRIATE SAVED REG BLOCK           COMCIDP     7765     A
COMCIDP            LX1    -18-3                                                          COMCIDP     7766     A
COMCIDP            BX6    X1                                                             COMCIDP     7767     A
COMCIDP            AX6    59          (X6) = +0 IF *J* REG + *K* REG                     COMCIDP     7768     A
COMCIDP  *                                  -0 IF *J* REG - *K* REG                      COMCIDP     7769     A
COMCIDP            SB7    X1                                                             COMCIDP     7770     A
COMCIDP            ZR     X2,STP5     IF *SA0* (I.E. NOT LOAD/STORE)                     COMCIDP     7771     A
COMCIDP            MI     B6,STP13    IF *Q* 18 BIT ADDR                                 COMCIDP     7772     A
COMCIDP            SA5    B6+X4       (X5) = SAVED *K* REGISTER                          COMCIDP     7773     A
COMCIDP            BX5    X5-X6                                                          COMCIDP     7774     A
COMCIDP                                                                                  COMCIDP     7775     A
COMCIDP   STP13    SA1    B7+X3       (X1) = SAVED *J* REGISTER                          COMCIDP     7776     A
COMCIDP            IX6    X1+X5                                                          COMCIDP     7777     A
COMCIDP            SB2    X6          (B2) = ADDR TO LOAD/STORE                          COMCIDP     7778     A
COMCIDP            SA6    IDPXAR                                                         COMCIDP     7779     A
COMCIDP            RJ     =XCHK       CHECK CM ADDRESS                                   COMCIDP     7780     A
COMCIDP            PL     B2,STP5     IF ADDR OK                                         COMCIDP     7781     A
COMCIDP            EQ     STP=ER                                                         COMCIDP     7782     A
COMCIDP                                                                                  COMCIDP     7783     A
COMCIDP  *         LOAD/STORE TABLE.                                                     COMCIDP     7784     A
COMCIDP                                                                                  COMCIDP     7785     A
COMCIDP   STP=LS   BSS    0                                                              COMCIDP     7786     A
COMCIDP                                                                                  COMCIDP     7787     A
COMCIDP            LOC    50B                                                            COMCIDP     7788     A
COMCIDP   +        VFD    18/0,3/0,18/IDP=SVA,3/0,18/-1         SAI  AJ+Q                COMCIDP     7789     A
COMCIDP   +        VFD    18/0,3/0,18/IDP=SVB,3/0,18/-1         SAI  BJ+Q                COMCIDP     7790     A
COMCIDP   +        VFD    18/0,3/0,18/IDP=SVX,3/0,18/-1         SAI  XJ+Q                COMCIDP     7791     A
COMCIDP   +        VFD    18/0,3/0,18/IDP=SVX,3/0,18/IDP=SVB     SAI  XJ+BK              COMCIDP     7792     A
COMCIDP   +        VFD    18/0,3/0,18/IDP=SVA,3/0,18/IDP=SVB     SAI  AJ+BK              COMCIDP     7793     A
COMCIDP   +        VFD    18/0,3/0,18/IDP=SVA,3/4,18/IDP=SVB     SAI  AJ-BK              COMCIDP     7794     A
COMCIDP   +        VFD    18/0,3/0,18/IDP=SVB,3/0,18/IDP=SVB     SAI  BJ+BK              COMCIDP     7795     A
COMCIDP   +        VFD    18/0,3/0,18/IDP=SVB,3/4,18/IDP=SVB     SAI  BJ-BK              COMCIDP     7796     A
COMCIDP            LOC    *O                                                             COMCIDP     7797     A
COMCIDP                                                                                  COMCIDP     7798     A
COMCIDP  *         HERE TO UPDATE POS COUNTER AND PSEUDO P REGISTER.                     COMCIDP     7799     A
COMCIDP                                                                                  COMCIDP     7800     A
COMCIDP   STP=X    SA1    STP=OP                                                         COMCIDP     7801     A
COMCIDP            SA2    IDPXFR                                                         COMCIDP     7802     A
COMCIDP            SA3    IDPPREG                                                        COMCIDP     7803     A
COMCIDP            SA4    IDPPOS                                                         COMCIDP     7804     A
COMCIDP            SA5    BC=BRAD                                                        COMCIDP     7805     A
COMCIDP            ZR     X2,STP=ER   IF BAD XFER, BAD TIMES...                          COMCIDP     7806     A
COMCIDP            SA1    DAZ=PS+X1   (X1) = INSTRUCTION SKELETON                        COMCIDP     7807     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            SB5    15                                                    COMCIDP    7808    A
COMCIDP            BX6    X2                                                    COMCIDP    7809    A
COMCIDP            SB4    X4                                                    COMCIDP    7810    A
COMCIDP            MX7    0                                                     COMCIDP    7811    A
COMCIDP            PL     X1,STP=X2   IF SHORT (15 BIT) INSTRUCTION             COMCIDP    7812    A
COMCIDP            SB5    B5+B5                                                 COMCIDP    7813    A
COMCIDP                                                                         COMCIDP    7814    A
COMCIDP   STP=X2   LX1    59-57                                                 COMCIDP    7815    A
COMCIDP            PL     X2,STP=X4A  IF A XFER OF CONTROL OCCURRED             COMCIDP    7816    A
COMCIDP            SX6    X3+B1                                                 COMCIDP    7817    A
COMCIDP            LE     B4,B5,STP=X3  IF NO MORE INSTRUCTIONS IN THIS WORD    COMCIDP    7818    A
COMCIDP            PL     X1,STP=X5   IF INSTRUCTION DOES NOT FORCE UPPER       COMCIDP    7819    A
COMCIDP                                                                         COMCIDP    7820    A
COMCIDP   STP=X3   ZR     X5,STP=X4   IF NOT IN IDP GENERATED BREAK MODE        COMCIDP    7821    A
COMCIDP            SX6    X3                                                    COMCIDP    7822    A
COMCIDP                                                                         COMCIDP    7823    A
COMCIDP   *        HERE IF INSTRUCTION FORCES UPPER.                            COMCIDP    7824    A
COMCIDP                                                                         COMCIDP    7825    A
COMCIDP   STP=X4   SA7    A5          SET TO *NOT IN IDP GENERATED BREAK MODE*  COMCIDP    7826    A
COMCIDP            SA6    A3                                                    COMCIDP    7827    A
COMCIDP            SX7    60                                                    COMCIDP    7828    A
COMCIDP            SA7    A4                                                    COMCIDP    7829    A
COMCIDP            EQ     STP=X6                                                COMCIDP    7830    A
COMCIDP                                                                         COMCIDP    7831    A
COMCIDP   *        HERE IF TRANSFER OF CONTROL OCCURRED -- ENTER TRANSFER ADDR   COMCIDP    7832    A
COMCIDP   *          ONTO PUSH DOWN STACK OF SAVED TRANSFER ADDRESSES *IDPXFT*.  COMCIDP    7833    A
COMCIDP   *          ALL ENTRIES IN PUSH DOWN STACK ARE MOVED UP (TOWARD *RA*)   COMCIDP    7834    A
COMCIDP   *          ONE POSITION (THE 1ST ENTRY IN *XFT*, THE EARLIEST TRANSFER COMCIDP    7835    A
COMCIDP   *          ADDRESS, FALLS OFF STACK).  EXAMPLE --                      COMCIDP    7836    A
COMCIDP   *                                                                     COMCIDP    7837    A
COMCIDP   *                   BEFORE                 AFTER                      COMCIDP    7838    A
COMCIDP   *            XFT+0  XFER(N)         XFT+0  XFER(N+1)                  COMCIDP    7839    A
COMCIDP   *               +1  XFER(N+1)          +1  XFER(N+2)                  COMCIDP    7840    A
COMCIDP   *               +2  XFER(N+2)          +2  XFER(N+3)                  COMCIDP    7841    A
COMCIDP   *               +3  XFER(N+3)          +3  XFER(N+4) -- NEW ENTRY     COMCIDP    7842    A
COMCIDP   *               +4  END-OF-TABLE       +4  END-OF-TABLE               COMCIDP    7843    A
COMCIDP                                                                         COMCIDP    7844    A
COMCIDP   STP=X4A  SA1    IDPXFT+1    (A1,X1) = A+C OF 1ST WORD TO MOVE         COMCIDP    7845    A
COMCIDP            SB7    L.XFT-1     (B7) = NR OF WORDS TO MOVE                COMCIDP    7846    A
COMCIDP                                                                         COMCIDP    7847    A
COMCIDP   STP=X4B  BX7    X1                                                    COMCIDP    7848    A
COMCIDP            SA7    A1-B1                                                 COMCIDP    7849    A
COMCIDP            SB7    B7-B1                                                 COMCIDP    7850    A
COMCIDP            SA1    A1+B1                                                 COMCIDP    7851    A
COMCIDP            GT     B7,B0,STP=X4B IF NOT DONE                             COMCIDP    7852    A
COMCIDP                                                                         COMCIDP    7853    A
COMCIDP            LX4    59-29                                                 COMCIDP    7854    A
COMCIDP            BX7    X4+X3       (X7) = 30/POS COUNTER,30/P REG OF XFER INST COMCIDP    7855    A
COMCIDP            SA7    A1-B1       MAKE NEW ENTRY                            COMCIDP    7856    A
COMCIDP            MX7    0                                                     COMCIDP    7857    A
COMCIDP            EQ     STP=X4      FORCE UPPER...                            COMCIDP    7858    A
COMCIDP                                                                         COMCIDP    7859    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *         HERE IF INSTRUCTION DOES NOT FORCE UPPER.                COMCIDP    7860    A
COMCIDP                                                                     COMCIDP    7861    A
COMCIDP   STP=X5   SB6    B4-B5                                             COMCIDP    7862    A
COMCIDP            SX7    B6                                                COMCIDP    7863    A
COMCIDP            SA7    A4                                                COMCIDP    7864    A
COMCIDP                                                                     COMCIDP    7865    A
COMCIDP  *         LIST RESULT REGISTER.                                    COMCIDP    7866    A
COMCIDP                                                                     COMCIDP    7867    A
COMCIDP   STP=X6   SA3    IDPXLST                                           COMCIDP    7868    A
COMCIDP            SA4    STP=OP                                            COMCIDP    7869    A
COMCIDP            ZR     X3,EXIT.    IF NOT LISTING                        COMCIDP    7870    A
COMCIDP                                                                     COMCIDP    7871    A
COMCIDP            SX6    X4-10B                                            COMCIDP    7872    A
COMCIDP            SA3    STP=I                                             COMCIDP    7873    A
COMCIDP            MI     X6,EXIT.    IF A BRANCH INSTRUCTION               COMCIDP    7874    A
COMCIDP            LX1    -2          RESTORE (X1) = INSTRUCTION SKELETON   COMCIDP    7875    A
COMCIDP            MX0    -CHAR                                             COMCIDP    7876    A
COMCIDP            BX6    -X0*X1                                            COMCIDP    7877    A
COMCIDP            SX7    X6-1RB                                            COMCIDP    7878    A
COMCIDP            SB6    000B        (B6) = 0TR, WHERE T=0 (B REG)         COMCIDP    7879    A
COMCIDP            ZR     X7,STP=X8   IF B REGISTER                         COMCIDP    7880    A
COMCIDP            SX7    X6-1RA                                            COMCIDP    7881    A
COMCIDP            SB6    010B        (B6) = 0TR, WHERE T=1 (A REG)         COMCIDP    7882    A
COMCIDP            ZR     X7,STP=X8   IF A REGISTER                         COMCIDP    7883    A
COMCIDP            SX7    X6-1RX                                            COMCIDP    7884    A
COMCIDP            SB6    020B        (B6) = 0TR, WHERE T=2 (X REG)         COMCIDP    7885    A
COMCIDP            ZR     X7,STP=X9   IF X REGISTER                         COMCIDP    7886    A
COMCIDP            EQ     EXIT.                                             COMCIDP    7887    A
COMCIDP                                                                     COMCIDP    7888    A
COMCIDP   STP=X8   SB4    B6+X3       (B4) = 0TR                            COMCIDP    7889    A
COMCIDP            RJ     DAB         DUMP -A- OR -B- REGISTER              COMCIDP    7890    A
COMCIDP            EQ     EXIT.                                             COMCIDP    7891    A
COMCIDP                                                                     COMCIDP    7892    A
COMCIDP   STP=X9   SB4    B6+X3       (B4) = 02R                            COMCIDP    7893    A
COMCIDP            RJ     DUX         DUMP -X- REGISTER                     COMCIDP    7894    A
COMCIDP            EQ     EXIT.                                             COMCIDP    7895    A
COMCIDP                                                                     COMCIDP    7896    A
COMCIDP  *         HERE IF AN ERROR ENCOUNTERED.                            COMCIDP    7897    A
COMCIDP                                                                     COMCIDP    7898    A
COMCIDP   STP=ER   SA1    IDPXLST                                           COMCIDP    7899    A
COMCIDP            SA2    BC=BRAD                                           COMCIDP    7900    A
COMCIDP            SA3    IDPPREG                                           COMCIDP    7901    A
COMCIDP            SA4    IDPPOS                                            COMCIDP    7902    A
COMCIDP            NZ     X1,STP=E3   IF INSTRUCTION WAS ALREADY LISTED     COMCIDP    7903    A
COMCIDP            SA5    X3                                                COMCIDP    7904    A
COMCIDP            ZR     X2,STP=E2   IF NOT IN IDP GENERATED BREAK MODE    COMCIDP    7905    A
COMCIDP            SA5    A2                                                COMCIDP    7906    A
COMCIDP                                                                     COMCIDP    7907    A
COMCIDP   STP=E2   SB4    X4                                                COMCIDP    7908    A
COMCIDP            RJ     DAZ         DEASSEMBLE BAD INSTRUCTION            COMCIDP    7909    A
COMCIDP                                                                     COMCIDP    7910    A
COMCIDP   STP=E3   MX0    -18                                               COMCIDP    7911    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            SA1    IDPPREG                                                COMCIDP    7912    A
COMCIDP            SA2    =10H      P =                                          COMCIDP    7913    A
COMCIDP            BX1    -X0*X1                                                 COMCIDP    7914    A
COMCIDP            LX7    X2                                                     COMCIDP    7915    A
COMCIDP            SA7    SNAPLNE                                                COMCIDP    7916    A
COMCIDP            RJ     =XCOD       CONVERT BINARY TO OCTAL DPC                COMCIDP    7917    A
COMCIDP            BX6    X4                                                     COMCIDP    7918    A
COMCIDP            SA6    A7+B1                                                  COMCIDP    7919    A
COMCIDP            SA1    IDPPREG                                                COMCIDP    7920    A
COMCIDP            RJ     FRA         FIND RELATIVE ADDRESS                      COMCIDP    7921    A
COMCIDP            SA6    A6+B1                                                  COMCIDP    7922    A
COMCIDP            SA7    A6+B1                                                  COMCIDP    7923    A
COMCIDP            PRIDP  SNAPLNE                                                COMCIDP    7924    A
COMCIDP            EQ     IDP=ER                                                 COMCIDP    7925    A
COMCIDP                                                                          COMCIDP    7926    A
COMCIDP                                                                          COMCIDP    7927    A
COMCIDP   STP=ADR  =      DAZ=ADR                                                COMCIDP    7928    A
COMCIDP   STP=I    =      DAZ=I                                                  COMCIDP    7929    A
COMCIDP   STP=J    =      DAZ=J                                                  COMCIDP    7930    A
COMCIDP   STP=K    =      DAZ=K                                                  COMCIDP    7931    A
COMCIDP   STP=OP   =      DAZ=OP                                                 COMCIDP    7932    A
COMCIDP   TOGEL    SPACE  4,10                                                   COMCIDP    7933    A
COMCIDP   ***      THE *TOGEL* PROGRAM.                                           COMCIDP    7934    A
COMCIDP   *                                                                      COMCIDP    7935    A
COMCIDP   *                                                                      COMCIDP    7936    A
COMCIDP   *        THE FOLLOWING SECTION CONTAINS THE *TOGEL* PROGRAM THAT       COMCIDP    7937    A
COMCIDP   *        DRIVE/DESCRIBES THE GENERATION OF *IDP* TOKENS.               COMCIDP    7938    A
COMCIDP   IDPTOM   SPACE  4,10                                                   COMCIDP    7939    A
COMCIDP   IDPTOM   TOGEL                                                         COMCIDP    7940    A
COMCIDP                                                                          COMCIDP    7941    A
COMCIDP   TOM=NXT  BSS    0           ** MAIN LOOP NODE **                       COMCIDP    7942    A
COMCIDP            GROUP (),NUL,SQZ                                              COMCIDP    7943    A
COMCIDP   EOS      IFT ("EOS")                                                   COMCIDP    7944    A
COMCIDP              CALT TOK=EOS     GENERATE *EOS* TOKEN                       COMCIDP    7945    A
COMCIDP                                                                          COMCIDP    7946    A
COMCIDP   TOM=BOL  BSS    0           ** BEGINNING OF LINE **                    COMCIDP    7947    A
COMCIDP              ON EOL,TOK=EOS                                              COMCIDP    7948    A
COMCIDP              GROUP (),NUL,SQZ                                            COMCIDP    7949    A
COMCIDP              IFT (-A..Z0..9"EOS"/55)                                     COMCIDP    7950    A
COMCIDP                CASEOF (+..:),TOM=COF                                     COMCIDP    7951    A
COMCIDP                GOTO TOM=BOL                                              COMCIDP    7952    A
COMCIDP              ENDT                                                        COMCIDP    7953    A
COMCIDP              GROUP (A..Z),KEY,NSQZ                                       COMCIDP    7954    A
COMCIDP              GOTO TOM=NXT                                                COMCIDP    7955    A
COMCIDP   EOS      ENDT                                                          COMCIDP    7956    A
COMCIDP                                                                          COMCIDP    7957    A
COMCIDP   VAR      IFT (A..Z)                                                    COMCIDP    7958    A
COMCIDP              ON EOL,TOK=VOL                                              COMCIDP    7959    A
COMCIDP              GROUP (A..Z0..9),VAR,SQZ                                    COMCIDP    7960    A
COMCIDP              CALT TOK=VAR     PROCESS/CHECK *VAR* TOKEN                  COMCIDP    7961    A
COMCIDP              ON EOL,TOK=EOS                                              COMCIDP    7962    A
COMCIDP   VAR      ELST                                                          COMCIDP    7963    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP   CON        IFT (0..9)                                          COMCIDP    7964    A
COMCIDP                GROUP (0..9),CONS,SQZ                             COMCIDP    7965    A
COMCIDP   CON        ELST                                                COMCIDP    7966    A
COMCIDP                CASEOF (+..:)                                     COMCIDP    7967    A
COMCIDP   TOM=COF        TOKEN PLUS   +                                  COMCIDP    7968    A
COMCIDP                  TOKEN MIN    -                                  COMCIDP    7969    A
COMCIDP                  TOKEN STAR   *                                  COMCIDP    7970    A
COMCIDP                  TOKEN ILL    /                                  COMCIDP    7971    A
COMCIDP                  TOKEN LP     (                                  COMCIDP    7972    A
COMCIDP                  TOKEN RP     )                                  COMCIDP    7973    A
COMCIDP                  TOKEN SHFT   $                                  COMCIDP    7974    A
COMCIDP                  TOKEN ILL    =                                  COMCIDP    7975    A
COMCIDP                  TOKEN ILL    BLANK                              COMCIDP    7976    A
COMCIDP                  TOKEN COM    ,                                  COMCIDP    7977    A
COMCIDP                  TOKEN PER    .                                  COMCIDP    7978    A
COMCIDP                    TOKEN AND,(PER,VAR'AND',PER)                  COMCIDP    7979    A
COMCIDP                    TOKEN AND,(PER,VAR'A',PER)                    COMCIDP    7980    A
COMCIDP                    TOKEN OR,(PER,VAR'OR',PER)                    COMCIDP    7981    A
COMCIDP                    TOKEN OR,(PER,VAR'O',PER)                     COMCIDP    7982    A
COMCIDP                    TOKEN XOR,(PER,VAR'XOR',PER)                  COMCIDP    7983    A
COMCIDP                    TOKEN XOR,(PER,VAR'X',PER)                    COMCIDP    7984    A
COMCIDP                    TOKEN NOT,(PER,VAR'NOT',PER)                  COMCIDP    7985    A
COMCIDP                    TOKEN NOT,(PER,VAR'N',PER)                    COMCIDP    7986    A
COMCIDP                  TOKEN ILL    #                                  COMCIDP    7987    A
COMCIDP                  TOKEN ILL    [                                  COMCIDP    7988    A
COMCIDP                  TOKEN ILL    ]                                  COMCIDP    7989    A
COMCIDP                  TOKEN ILL    %                                  COMCIDP    7990    A
COMCIDP                  TOKEN ILL    DQOT                               COMCIDP    7991    A
COMCIDP                  TOKEN ILL    _                                  COMCIDP    7992    A
COMCIDP                  TOKEN ILL    !                                  COMCIDP    7993    A
COMCIDP                  TOKEN ILL    &                                  COMCIDP    7994    A
COMCIDP                  TOKEN SQOT,,TOK=SQT    '                        COMCIDP    7995    A
COMCIDP                  TOKEN QQQ    ?                                  COMCIDP    7996    A
COMCIDP                  TOKEN ILL    <                                  COMCIDP    7997    A
COMCIDP                  TOKEN ILL    >                                  COMCIDP    7998    A
COMCIDP                  TOKEN ILL    @                                  COMCIDP    7999    A
COMCIDP                  TOKEN ILL    \                                  COMCIDP    8000    A
COMCIDP                  TOKEN ILL    ^                                  COMCIDP    8001    A
COMCIDP                  TOKEN ILL    SCOL                               COMCIDP    8002    A
COMCIDP                  TOKEN ILL    :                                  COMCIDP    8003    A
COMCIDP                ENDC                                              COMCIDP    8004    A
COMCIDP   CON        ENDT                                                COMCIDP    8005    A
COMCIDP   VAR      ENDT                                                  COMCIDP    8006    A
COMCIDP            GOTO TOM=NXT                                          COMCIDP    8007    A
COMCIDP   TOM=SQT  SPACE  4,10                                           COMCIDP    8008    A
COMCIDP  **        TOM=SQT - HERE FOR SPECIAL 'XXX' *VAR* TOKEN.         COMCIDP    8009    A
COMCIDP                                                                  COMCIDP    8010    A
COMCIDP                                                                  COMCIDP    8011    A
COMCIDP   TOM=SQT  BSS    0          ** ENTRY **                         COMCIDP    8012    A
COMCIDP            GROUP  (-'"EOS"),VAR,SQZ                              COMCIDP    8013    A
COMCIDP            CALT TOK=SQV       CHECK SPECIAL 'VAR' TOKEN          COMCIDP    8014    A
COMCIDP            GOTO TOM=NXT                                          COMCIDP    8015    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP   IDPTOM   TOGEL                                                  COMCIDP     8016     A
COMCIDP   TOK=     SPACE  4,10                                            COMCIDP     8017     A
COMCIDP  ***       TOK= - IDP OWNCODE FOR *COMCTOK*.                      COMCIDP     8018     A
COMCIDP  *                                                                COMCIDP     8019     A
COMCIDP  *                                                                COMCIDP     8020     A
COMCIDP   TOK=EOS  SPACE  4,10                                            COMCIDP     8021     A
COMCIDP  **        TOK=EOS - END-OF-STMT PROCESSING.                      COMCIDP     8022     A
COMCIDP                                                                   COMCIDP     8023     A
COMCIDP                                                                   COMCIDP     8024     A
COMCIDP   TOK=EOS  BSS    0           ** ENTRY **                         COMCIDP     8025     A
COMCIDP            SA1    TB=STMT                                         COMCIDP     8026     A
COMCIDP            MX6    0                                               COMCIDP     8027     A
COMCIDP            SX2    A6+B1                                           COMCIDP     8028     A
COMCIDP            SA6    A6+B1       MARK EOS                            COMCIDP     8029     A
COMCIDP            IX7    X2-X1       (X7) = NR OF WORDS IN STMT JUST ENTOKENED  COMCIDP  8030  A
COMCIDP            LX7    TB.LENP-0                                       COMCIDP     8031     A
COMCIDP            SA7    X1                                              COMCIDP     8032     A
COMCIDP                                                                   COMCIDP     8033     A
COMCIDP            SX7    A6+B1       (X7) = ADDR OF LEN WORD FOR NEXT/NEW STMT  COMCIDP  8034  A
COMCIDP            SA6    A6+B1       MARK BOS                            COMCIDP     8035     A
COMCIDP            SA7    A1                                              COMCIDP     8036     A
COMCIDP            MX4    -1          SET TO *NO USEABLE CHAR IN (X4)*    COMCIDP     8037     A
COMCIDP            LE     B7,B0,=XTOK IF HERE FROM *ON EOL* PROCESSING, EXIT...  COMCIDP  8038  A
COMCIDP            EQ     =XTOK=MN                                        COMCIDP     8039     A
COMCIDP   TOK=ER   SPACE  4,10                                            COMCIDP     8040     A
COMCIDP  **        TOK=ER - HERE IF ERROR DURING TOKEN GENERATION.        COMCIDP     8041     A
COMCIDP  *                                                                COMCIDP     8042     A
COMCIDP  *                                                                COMCIDP     8043     A
COMCIDP  *         NEED TO CLEAR TOKEN BUFFER, *IDPTB*.                   COMCIDP     8044     A
COMCIDP                                                                   COMCIDP     8045     A
COMCIDP                                                                   COMCIDP     8046     A
COMCIDP   TOK=ER   BSS    0           ** ENTRY **                         COMCIDP     8047     A
COMCIDP            SX6    B1                                              COMCIDP     8048     A
COMCIDP            BX7    X7-X7                                           COMCIDP     8049     A
COMCIDP            LX6    TB.LENP-0                                       COMCIDP     8050     A
COMCIDP            SA6    IDPTB                                           COMCIDP     8051     A
COMCIDP            SA7    A6+B1       MARK EOS FOR NULL STMT/LINE         COMCIDP     8052     A
COMCIDP            SA7    A7+B1       MARK EOL FOR NULL STMT/LINE         COMCIDP     8053     A
COMCIDP            EQ     SE.TOK                                          COMCIDP     8054     A
COMCIDP   TOK=SQT  SPACE  4,10                                            COMCIDP     8055     A
COMCIDP  **        TOK=SQT - HERE IF ' ENCOUNTERED.                       COMCIDP     8056     A
COMCIDP  *                                                                COMCIDP     8057     A
COMCIDP  *                                                                COMCIDP     8058     A
COMCIDP                                                                   COMCIDP     8059     A
COMCIDP                                                                   COMCIDP     8060     A
COMCIDP   TOK=SQT  BSS    0           ** ENTRY **                         COMCIDP     8061     A
COMCIDP            SA0    TOM=SQT                                         COMCIDP     8062     A
COMCIDP            EQ     =XTOK=CON   RETURN W/ *NO STORE*...             COMCIDP     8063     A
COMCIDP   TOK=SQV  SPACE  4,10                                            COMCIDP     8064     A
COMCIDP  **        TOK=SQV - CHECK SPECIAL 'VAR' TOKEN.                   COMCIDP     8065     A
COMCIDP  *                                                                COMCIDP     8066     A
COMCIDP  *                                                                COMCIDP     8067     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP                                                                                COMCIDP     8068     A
COMCIDP                                                                                COMCIDP     8069     A
COMCIDP   TOK=SQV  BSS    0           ** ENTRY **                                      COMCIDP     8070     A
COMCIDP            SA1    A6-B1       (A1,X1) = A+C OF TOKEN PRECEDING *VAR*           COMCIDP     8071     A
COMCIDP            MX4    -1          SET TO *NO USEABLE CHAR IN (X4)*                 COMCIDP     8072     A
COMCIDP            SX7    X1-O.VAR                                                     COMCIDP     8073     A
COMCIDP            ZR     X7,TOK=ER   IF *VAR* TOKEN TOO LONG                          COMCIDP     8074     A
COMCIDP            EQ     =XTOK=MN                                                     COMCIDP     8075     A
COMCIDP   TOK=VAR  SPACE  4,10                                                         COMCIDP     8076     A
COMCIDP  **        TOK=VAR - PROCESS/CHECK *VAR* TOKEN.                                COMCIDP     8077     A
COMCIDP  *                                                                             COMCIDP     8078     A
COMCIDP  *                                                                             COMCIDP     8079     A
COMCIDP  *         CALL *VAR* (PROCESS *VAR* TOKEN) TO PERFORM THE FOLLOWING --        COMCIDP     8080     A
COMCIDP  *                                                                             COMCIDP     8081     A
COMCIDP  *           1. CHECK TO MAKE SURE THAT *VAR* TOKEN IS ONLY 1 TOKEN LONG,      COMCIDP     8082     A
COMCIDP  *              I.E. NR CHARS .LE. 7.                                          COMCIDP     8083     A
COMCIDP  *                                                                             COMCIDP     8084     A
COMCIDP  *           2. IF THIS *VAR* TOKEN IS A REGISTER NAME, CHANGE IT TO           COMCIDP     8085     A
COMCIDP  *              A *REG* TOKEN.                                                 COMCIDP     8086     A
COMCIDP                                                                                COMCIDP     8087     A
COMCIDP                                                                                COMCIDP     8088     A
COMCIDP   TOK=VAR  BSS    0           ** ENTRY **                                      COMCIDP     8089     A
COMCIDP            SA1    A6          (A1,X1) = A+C OF *VAR* TOKEN                     COMCIDP     8090     A
COMCIDP            RJ     VAR         PROCESS *VAR* TOKEN                              COMCIDP     8091     A
COMCIDP            EQ     =XTOK=MN                                                     COMCIDP     8092     A
COMCIDP   TOK=VOL  SPACE  4,10                                                         COMCIDP     8093     A
COMCIDP  **        TOK=VOL - HERE IF *VAR* TOKEN TERMINATED BY *EOL*.                  COMCIDP     8094     A
COMCIDP  *                                                                             COMCIDP     8095     A
COMCIDP  *                                                                             COMCIDP     8096     A
COMCIDP  *         HERE VIA *ON EOL,TOK=VOL*.  NEED TO DO THE SAME THING               COMCIDP     8097     A
COMCIDP  *         THAT *TOK=VAR* DOES, **EXCEPT** THAT OUR POINTERS ARE               COMCIDP     8098     A
COMCIDP  *         A LITTLE DIFFERENT.                                                 COMCIDP     8099     A
COMCIDP                                                                                COMCIDP     8100     A
COMCIDP                                                                                COMCIDP     8101     A
COMCIDP   TOK=VOL  BSS    0           ** ENTRY **                                      COMCIDP     8102     A
COMCIDP            SA1    A6-B1       (A1,X1) = A+C OF *VAR* TOKEN                     COMCIDP     8103     A
COMCIDP            RJ     VAR         PROCESS *VAR* TOKEN                              COMCIDP     8104     A
COMCIDP            EQ     TOK=EOS                                                      COMCIDP     8105     A
COMCIDP   UBK      SPACE  4,10                                                         COMCIDP     8106     A
COMCIDP  **        UBK - UNBREAK BREAKS WITHIN A RANGE.                                COMCIDP     8107     A
COMCIDP  *                                                                             COMCIDP     8108     A
COMCIDP  *                                                                             COMCIDP     8109     A
COMCIDP  *         ENTRY  (B2) = LOWER BOUND TO UNBREAK.                               COMCIDP     8110     A
COMCIDP  *                (B3) = UPPER BOUND TO UNBREAK.                               COMCIDP     8111     A
COMCIDP  *                                                                             COMCIDP     8112     A
COMCIDP  *                     I.E. *UBK* WILL UNBREAK ALL BREAKS THAT FALL            COMCIDP     8113     A
COMCIDP  *                     IN THE RANGE --                                         COMCIDP     8114     A
COMCIDP  *                                                                             COMCIDP     8115     A
COMCIDP  *                          ADDR .GE. (B2) .AND. ADDR .LE. (B3)                COMCIDP     8116     A
COMCIDP  *                                                                             COMCIDP     8117     A
COMCIDP  *         EXIT   NONE                                                         COMCIDP     8118     A
COMCIDP  *                                                                             COMCIDP     8119     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *         USES   X - 1,2,3,4,6,7                                    COMCIDP    8120    A
COMCIDP  *                A - 1,2,3,4,6,7                                    COMCIDP    8121    A
COMCIDP  *                B - 4,6,7                                          COMCIDP    8122    A
COMCIDP  *                                                                   COMCIDP    8123    A
COMCIDP  *         CALLS  NONE                                               COMCIDP    8124    A
COMCIDP                                                                      COMCIDP    8125    A
COMCIDP                                                                      COMCIDP    8126    A
COMCIDP   UBK      SUBR                ** ENTRY/EXIT **                       COMCIDP    8127    A
COMCIDP            SB7    -1                                                 COMCIDP    8128    A
COMCIDP                                                                      COMCIDP    8129    A
COMCIDP   UBK2     SB7    B7+1                                               COMCIDP    8130    A
COMCIDP            SA1    IDPBA+B7                                           COMCIDP    8131    A
COMCIDP                                                                      COMCIDP    8132    A
COMCIDP            ZR     X1,EXIT.    IF END-OF-TABLE, DONE...               COMCIDP    8133    A
COMCIDP            SX6    X1-1                                               COMCIDP    8134    A
COMCIDP            ZR     X6,UBK2     IF AVAILABLE (EMPTY) ENTRY, IGNORE...  COMCIDP    8135    A
COMCIDP                                                                      COMCIDP    8136    A
COMCIDP            SB4    X1+         (B4) = BREAK ADDR                      COMCIDP    8137    A
COMCIDP            LT     B4,B2,UBK2  IF BREAK ADDR TOO LOW                  COMCIDP    8138    A
COMCIDP            GT     B4,B3,UBK2  IF BREAK ADDR TOO HIGH                 COMCIDP    8139    A
COMCIDP                                                                      COMCIDP    8140    A
COMCIDP            SB6    B7+B7       (B6) = ORD INTO *IDPBC* FOR THIS BREAK COMCIDP    8141    A
COMCIDP            SA2    IDPBC+B6    (A2,X2) = A+C OF *IDPBC* ENTRY FOR THIS BRK COMCIDP 8142 A
COMCIDP            SA3    B4          (X3) = BREAKPOINT (I.E. SHOULD BE *RJ IDP*) COMCIDP 8143 A
COMCIDP            SA4    ST=BRKA     (X4) = *RJ IDP* PLUG FROM *BREAK*      COMCIDP    8144    A
COMCIDP            MX6    -1                                                 COMCIDP    8145    A
COMCIDP            BX7    X2                                                 COMCIDP    8146    A
COMCIDP            IX3    X3-X4                                              COMCIDP    8147    A
COMCIDP            SA6    A1          SET TO *THIS ENTRY AVAILABLE*          COMCIDP    8148    A
COMCIDP            NZ     X3,UBK2     IF BREAK ADDR DOESNT HAVE *RJ IDP*     COMCIDP    8149    A
COMCIDP            SA7    B4          RESTORE PREVIOUS CONTENTS OF BREAK ADDR COMCIDP    8150    A
COMCIDP            EQ     UBK2                                               COMCIDP    8151    A
COMCIDP   UFO      SPACE  4,10                                               COMCIDP    8152    A
COMCIDP  **        UFO - USER FREEZE OWNCODE.                                COMCIDP    8153    A
COMCIDP  *                                                                   COMCIDP    8154    A
COMCIDP  *                                                                   COMCIDP    8155    A
COMCIDP  *         *UFO* WILL CALL *UFO=* IF THE IDP INSTALLER/USER          COMCIDP    8156    A
COMCIDP  *         PROVIDED *UFO=* TO ALLOW THE HOST TO PERFORM ITS          COMCIDP    8157    A
COMCIDP  *         OWNCODE RIGHT AFTER IDP *FREEZE*S AN INTERACTIVE          COMCIDP    8158    A
COMCIDP  *         SESSION.                                                  COMCIDP    8159    A
COMCIDP  *                                                                   COMCIDP    8160    A
COMCIDP  *         ENTRY  NONE                                               COMCIDP    8161    A
COMCIDP  *                                                                   COMCIDP    8162    A
COMCIDP  *         EXIT   NONE                                               COMCIDP    8163    A
COMCIDP  *                                                                   COMCIDP    8164    A
COMCIDP  *         USES   ALL                                                COMCIDP    8165    A
COMCIDP  *                                                                   COMCIDP    8166    A
COMCIDP  *         CALLS  UFO=(IF AVAIL)                                     COMCIDP    8167    A
COMCIDP                                                                      COMCIDP    8168    A
COMCIDP                                                                      COMCIDP    8169    A
COMCIDP   UFO      SUBR                ** ENTRY/EXIT **                       COMCIDP    8170    A
COMCIDP            SB2    =YUFO=                                             COMCIDP    8171    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            MI     B2,EXIT.    IF USER DID NOT PROVIDE *UFO=*                COMCIDP    8172    A
COMCIDP            RJ     =YUFO=      USER FREEZE OWNCODE                           COMCIDP    8173    A
COMCIDP            EQ     EXIT.                                                     COMCIDP    8174    A
COMCIDP   UFR      SPACE  4,10                                                      COMCIDP    8175    A
COMCIDP  **        UFR - USER FREEZE RESTART OWNCODE.                               COMCIDP    8176    A
COMCIDP  *                                                                          COMCIDP    8177    A
COMCIDP  *                                                                          COMCIDP    8178    A
COMCIDP  *         *UFR* WILL CALL *UFR=* IF THE INSTALLER/USER                     COMCIDP    8179    A
COMCIDP  *         PROVIDED *UFR=* TO ALLOW THE HOST TO PERFORM ITS                 COMCIDP    8180    A
COMCIDP  *         OWNCODE RIGHT AFTER IDP RESTARTS A FROZEN INTERACTIVE            COMCIDP    8181    A
COMCIDP  *         SESSION.                                                         COMCIDP    8182    A
COMCIDP  *                                                                          COMCIDP    8183    A
COMCIDP  *         ENTRY  NONE                                                      COMCIDP    8184    A
COMCIDP  *                                                                          COMCIDP    8185    A
COMCIDP  *         EXIT   NONE                                                      COMCIDP    8186    A
COMCIDP  *                                                                          COMCIDP    8187    A
COMCIDP  *         USES   ALL                                                       COMCIDP    8188    A
COMCIDP  *                                                                          COMCIDP    8189    A
COMCIDP  *         CALLS  UFR=(IF AVAIL)                                            COMCIDP    8190    A
COMCIDP                                                                             COMCIDP    8191    A
COMCIDP                                                                             COMCIDP    8192    A
COMCIDP   UFR      SUBR               ** ENTRY/EXIT **                              COMCIDP    8193    A
COMCIDP            SB2    =YUFR=                                                    COMCIDP    8194    A
COMCIDP            MI     B2,EXIT.    IF USER DID NOT PROVIDE *UFR=*                COMCIDP    8195    A
COMCIDP            RJ     =YUFR=      USER FREEZE RESTART OWNCODE                   COMCIDP    8196    A
COMCIDP            EQ     EXIT.                                                     COMCIDP    8197    A
COMCIDP   UIO      SPACE  4,10                                                      COMCIDP    8198    A
COMCIDP  **        UIO - USER IDP OWNCODE.                                          COMCIDP    8199    A
COMCIDP  *                                                                          COMCIDP    8200    A
COMCIDP  *                                                                          COMCIDP    8201    A
COMCIDP  *         *UIO* WILL CALL *UIO=* TO ALLOW THE HOST TO PERFORM              COMCIDP    8202    A
COMCIDP  *         ITS OWNCODE IF THE IDP INSTALLER PROVIDED *UIO=*.                COMCIDP    8203    A
COMCIDP  *                                                                          COMCIDP    8204    A
COMCIDP  *         ENTRY  (FW=PARM) = FWA OF PARAMETER LIST FOR THIS *BREAK*.       COMCIDP    8205    A
COMCIDP  *                                                                          COMCIDP    8206    A
COMCIDP  *         EXIT   (X1) = .MI. IF BREAK IS TO BE HONORED, ELSE .PL.          COMCIDP    8207    A
COMCIDP  *                (X5) = FWA OF *BREAK* PARAMETER LIST.                     COMCIDP    8208    A
COMCIDP  *                                                                          COMCIDP    8209    A
COMCIDP  *         USES   ALL                                                       COMCIDP    8210    A
COMCIDP  *                                                                          COMCIDP    8211    A
COMCIDP  *         CALLS  UIO=(IF AVAIL)                                            COMCIDP    8212    A
COMCIDP                                                                             COMCIDP    8213    A
COMCIDP                                                                             COMCIDP    8214    A
COMCIDP   UIO      SUBR               ** ENTRY/EXIT **                              COMCIDP    8215    A
COMCIDP            SX1    =YUIO=                                                    COMCIDP    8216    A
COMCIDP            SA5    FW=PARM     (X5) = FWA OF PARAMETER LIST                  COMCIDP    8217    A
COMCIDP            MI     X1,EXIT.    IF USER DID NOT PROVIDE *UIO=*                COMCIDP    8218    A
COMCIDP            RJ     =YUIO=      USER IDP OWNCODE                              COMCIDP    8219    A
COMCIDP            SA5    FW=PARM     RESTORE (X5)                                  COMCIDP    8220    A
COMCIDP            EQ     EXIT.                                                     COMCIDP    8221    A
COMCIDP   .IDPOS   ENDIF                                                            CCGA018      38    A
COMCIDP   URO      SPACE  4,10                                                      COMCIDP    8222    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  **        URO - USER *REG* OWNCODE.                                COMCIDP    8223    A
COMCIDP  *                                                                  COMCIDP    8224    A
COMCIDP  *                                                                  COMCIDP    8225    A
COMCIDP  *         *URO* WILL CALL *URO=* TO ALLOW THE HOST TO PERFORM      COMCIDP    8226    A
COMCIDP  *         ITS OWNCODE IF THE IDP INSTALLER PROVIDED *URO=*.        COMCIDP    8227    A
COMCIDP  *                                                                  COMCIDP    8228    A
COMCIDP  *         ENTRY  (FW=PARM) = FWA OF PARAMETER LIST FOR THIS *REG*. COMCIDP    8229    A
COMCIDP  *                                                                  COMCIDP    8230    A
COMCIDP  *         EXIT   (X1) = .MI. IF REG IS TO BE HONORED, ELSE .PL.    COMCIDP    8231    A
COMCIDP  *                (X5) = FWA OF *REG* PARAMETER LIST.               COMCIDP    8232    A
COMCIDP  *                                                                  COMCIDP    8233    A
COMCIDP  *         USES   ALL                                               COMCIDP    8234    A
COMCIDP  *                                                                  COMCIDP    8235    A
COMCIDP  *         CALLS  URO=(IF AVAIL)                                    COMCIDP    8236    A
COMCIDP                                                                     COMCIDP    8237    A
COMCIDP                                                                     COMCIDP    8238    A
COMCIDP   URO      SUBR                ** ENTRY/EXIT **                     COMCIDP    8239    A
COMCIDP            SX1    =YURO=                                            COMCIDP    8240    A
COMCIDP            SA5    FW=PARM     (X5) = FWA OF PARAMETER LIST          COMCIDP    8241    A
COMCIDP            MI     X1,EXIT.    IF USER DID NOT PROVIDE *URO=*        COMCIDP    8242    A
COMCIDP            RJ     =YURO=      USER *REG* OWNCODE                    COMCIDP    8243    A
COMCIDP            SA5    FW=PARM     RESTORE (X5)                          COMCIDP    8244    A
COMCIDP            EQ     EXIT.                                             COMCIDP    8245    A
COMCIDP   USO      SPACE  4,10                                              COMCIDP    8246    A
COMCIDP  **        USO - USER *SNP* OWNCODE.                                COMCIDP    8247    A
COMCIDP  *                                                                  COMCIDP    8248    A
COMCIDP  *                                                                  COMCIDP    8249    A
COMCIDP  *         *USO* WILL CALL *USO=* TO ALLOW THE HOST TO PERFORM      COMCIDP    8250    A
COMCIDP  *         ITS OWNCODE IF THE IDP INSTALLER PROVIDED *USO=*.        COMCIDP    8251    A
COMCIDP  *                                                                  COMCIDP    8252    A
COMCIDP  *         ENTRY  (FW=PARM) = FWA OF PARAMETER LIST FOR THIS *SNAP*. COMCIDP    8253    A
COMCIDP  *                                                                  COMCIDP    8254    A
COMCIDP  *         EXIT   (X1) = .MI. IF SNAP IS TO BE HONORED, ELSE .PL.   COMCIDP    8255    A
COMCIDP  *                (X5) = FWA OF *SNAP* PARAMETER LIST.              COMCIDP    8256    A
COMCIDP  *                                                                  COMCIDP    8257    A
COMCIDP  *         USES   ALL                                               COMCIDP    8258    A
COMCIDP  *                                                                  COMCIDP    8259    A
COMCIDP  *         CALLS  USO=(IF AVAIL)                                    COMCIDP    8260    A
COMCIDP                                                                     COMCIDP    8261    A
COMCIDP                                                                     COMCIDP    8262    A
COMCIDP   USO      SUBR                ** ENTRY/EXIT **                     COMCIDP    8263    A
COMCIDP            SX1    =YUSO=                                            COMCIDP    8264    A
COMCIDP            SA5    FW=PARM     (X5) = FWA OF PARAMETER LIST          COMCIDP    8265    A
COMCIDP            MI     X1,EXIT.    IF USER DID NOT PROVIDE *USO=*        COMCIDP    8266    A
COMCIDP            RJ     =YUSO=      USER *SNP* OWNCODE                    COMCIDP    8267    A
COMCIDP            SA5    FW=PARM     RESTORE (X5)                          COMCIDP    8268    A
COMCIDP            EQ     EXIT.                                             COMCIDP    8269    A
COMCIDP   VAR      SPACE  4,10                                              COMCIDP    8270    A
COMCIDP   .IDPOS   IFNE   .OS,2                                             CCGA018      39    A
COMCIDP  **        VAR - PROCESS *VAR* TOKEN.                               COMCIDP    8271    A
COMCIDP  *                                                                  COMCIDP    8272    A
COMCIDP  *                                                                  COMCIDP    8273    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP  *         THIS ROUTINE PERFORMS SOME SPECIAL CASE PROCESSING                COMCIDP    8274    A
COMCIDP  *         UPON A VARIABLE NAME, *VAR*, TOKEN DURING TOKEN                   COMCIDP    8275    A
COMCIDP  *         GENERATION --                                                     COMCIDP    8276    A
COMCIDP  *                                                                           COMCIDP    8277    A
COMCIDP  *           1. *VAR* MAKES SURE THAT THIS *VAR* TOKEN IS NOT                COMCIDP    8278    A
COMCIDP  *              TOO LONG, I.E. NR CHARS .LE. 7.  THIS IS NECESSARY           COMCIDP    8279    A
COMCIDP  *              FOR 2 PRIMARY REASONS:  FIRST, IDP IS NOT SET UP             COMCIDP    8280    A
COMCIDP  *              TO HANDLE A *VAR* TOKEN THAT IS LONGER THAN A SINGLE         COMCIDP    8281    A
COMCIDP  *              CM WORD.  CHECKING FOR THIS CONDITION EARLY (I.E. DURING     COMCIDP    8282    A
COMCIDP  *              TOKEN GENERATION) SHOULD SIMPLIFY THE WORK OF OTHERS.        COMCIDP    8283    A
COMCIDP  *                                                                           COMCIDP    8284    A
COMCIDP  *              SECOND, BECAUSE *COMCTOK* CAN NOT BE TOLD TO GENERATE        COMCIDP    8285    A
COMCIDP  *              A *VAR* TOKEN THAT IS 1 CM WORD LONG, WE HAVE TO CHECK       COMCIDP    8286    A
COMCIDP  *              OURSELVES.                                                   COMCIDP    8287    A
COMCIDP  *                                                                           COMCIDP    8288    A
COMCIDP  *           2. BECAUSE HARDWARE REGISTER NAMES **LOOK** LIKE                COMCIDP    8289    A
COMCIDP  *              VARIABLE NAMES, IT IS THE RESPONSIBILITY OF *VAR* TO         COMCIDP    8290    A
COMCIDP  *              CHECK FOR, AND REPLACE REGISTER NAMES WITH AN APPROPRIATE    COMCIDP    8291    A
COMCIDP  *              *REG* TOKEN.                                                 COMCIDP    8292    A
COMCIDP  *                                                                           COMCIDP    8293    A
COMCIDP  *         ENTRY  (A1,X1) = A+C OF *VAR* TOKEN TO CHECK.                     COMCIDP    8294    A
COMCIDP  *                                                                           COMCIDP    8295    A
COMCIDP  *         EXIT   (A6)+1  = ADDR TO STORE NEXT TOKEN.                        COMCIDP    8296    A
COMCIDP  *                                                                           COMCIDP    8297    A
COMCIDP  *                TO *TOK=ER* IF *VAR* TOKEN TOO LONG.                       COMCIDP    8298    A
COMCIDP  *                                                                           COMCIDP    8299    A
COMCIDP  *         USES   X - 1,2,3,6,7                                              COMCIDP    8300    A
COMCIDP  *                A - 2,3,6                                                  COMCIDP    8301    A
COMCIDP  *                B - 2                                                      COMCIDP    8302    A
COMCIDP  *                                                                           COMCIDP    8303    A
COMCIDP  *         CALLS  TOK=ER                                                     COMCIDP    8304    A
COMCIDP                                                                              COMCIDP    8305    A
COMCIDP                                                                              COMCIDP    8306    A
COMCIDP   VAR      SUBR                ** ENTRY/EXIT **                              COMCIDP    8307    A
COMCIDP            SA3    A1-B1        (A3,X3) = A+C OF TOKEN PRECEDING *VAR*        COMCIDP    8308    A
COMCIDP            SA2    MX=BAX       (X2) = *BAX* CHARACTER SHIFT MASK             COMCIDP    8309    A
COMCIDP            LX1    CHAR                                                       COMCIDP    8310    A
COMCIDP            SX7    X3-O.VAR                                                   COMCIDP    8311    A
COMCIDP            ZR     X7,TOK=ER    IF VARIABLE NAME IS TOO LONG, ERROR...        COMCIDP    8312    A
COMCIDP                                                                              COMCIDP    8313    A
COMCIDP  *         CHECK FOR REGISTER NAME.                                          COMCIDP    8314    A
COMCIDP                                                                              COMCIDP    8315    A
COMCIDP            BX6    -X0*X1                                                     COMCIDP    8316    A
COMCIDP            LX1    CHAR                                                       COMCIDP    8317    A
COMCIDP            SB2    X6                                                         COMCIDP    8318    A
COMCIDP            LX2    B2                                                         COMCIDP    8319    A
COMCIDP            PL     X2,EXIT.     IF 1ST CHAR IS NOT *BAX*                      COMCIDP    8320    A
COMCIDP            SX7    020B         (X7) = 0TR, FOR -X- REG                       COMCIDP    8321    A
COMCIDP            SB2    X6-1RX                                                     COMCIDP    8322    A
COMCIDP            ZR     B2,VAR2      IF -X- REG                                    COMCIDP    8323    A
COMCIDP            SB2    X6-1RB                                                     COMCIDP    8324    A
COMCIDP            SX7    000B         (X7) = 0TR, FOR -B- REG                       COMCIDP    8325    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP            ZR     B2,VAR2     IF -B- REG                                    COMCIDP    8326    A
COMCIDP            SX7    010B        (X7) = 0TR, FOR -A- REG                       COMCIDP    8327    A
COMCIDP                                                                             COMCIDP    8328    A
COMCIDP   VAR2     SA2    MX=0..7     (X2) = *01234567* CHAR SHIFT MASK             COMCIDP    8329    A
COMCIDP            BX6    -X0*X1                                                    COMCIDP    8330    A
COMCIDP            LX1    CHAR                                                      COMCIDP    8331    A
COMCIDP            SB2    X6                                                        COMCIDP    8332    A
COMCIDP            LX2    B2                                                        COMCIDP    8333    A
COMCIDP            PL     X2,EXIT.    IF 2ND CHAR IS NOT *0..7*                     COMCIDP    8334    A
COMCIDP            SX6    X6-1R0                                                    COMCIDP    8335    A
COMCIDP            MX2    5*CHAR                                                    COMCIDP    8336    A
COMCIDP            IX3    X7+X6       (X7) = 0TR                                    COMCIDP    8337    A
COMCIDP            BX6    X2*X1                                                     COMCIDP    8338    A
COMCIDP            LX3    TB.0TRP-0                                                 COMCIDP    8339    A
COMCIDP            NZ     X6,EXIT.    IF 3RD THRU 7TH CHARS NON-NULL                COMCIDP    8340    A
COMCIDP            SX1    O.REG                                                     COMCIDP    8341    A
COMCIDP            BX6    X3+X1       MERGE *0TR* AND *TOT*                         COMCIDP    8342    A
COMCIDP            SA6    A1                                                        COMCIDP    8343    A
COMCIDP            EQ     EXIT.                                                     COMCIDP    8344    A
COMCIDP   VFD      SPACE  4,10                                                      COMCIDP    8345    A
COMCIDP  **        VFD - VARIABLE FIELD DEFINITION.                                 COMCIDP    8346    A
COMCIDP  *                                                                          COMCIDP    8347    A
COMCIDP  *                                                                          COMCIDP    8348    A
COMCIDP  *                CONVERTS A SPECIFIED FIELD IN A BINARY WORD TO OCTAL      COMCIDP    8349    A
COMCIDP  *         DISPLAY CODE WITH APPROPRIATE SPACING AND BLANK FILL.            COMCIDP    8350    A
COMCIDP  *         THE FIELD TO BE CONVERTED CAN BE REPRESENTED IN *COMPASS* BY--   COMCIDP    8351    A
COMCIDP  *                                                                          COMCIDP    8352    A
COMCIDP  *         POS    (B4)   IN (X1)                                            COMCIDP    8353    A
COMCIDP  *         VFD    (B5)/FIELD   WHERE *FIELD* HAS BEEN EXTRACTED FROM (X1)   COMCIDP    8354    A
COMCIDP  *                                                                          COMCIDP    8355    A
COMCIDP  *         E.G.   (B4) = 45   POSITION COUNTER                              COMCIDP    8356    A
COMCIDP  *                (B5) = 30   NR OF BITS                                    COMCIDP    8357    A
COMCIDP  *                                                                          COMCIDP    8358    A
COMCIDP  *                BINARY      76211031100015620310                          COMCIDP    8359    A
COMCIDP  *                DPC RSULT   .....0311000156.....    (.=BLANK(55B))        COMCIDP    8360    A
COMCIDP  *                                                                          COMCIDP    8361    A
COMCIDP  *         ENTRY  (X1)  =  BINARY WORD TO BE CONVERTED                      COMCIDP    8362    A
COMCIDP  *                (B4)  =  POSITION COUNTER                                 COMCIDP    8363    A
COMCIDP  *                (B5)  =  NR OF BITS IN FIELD                              COMCIDP    8364    A
COMCIDP  *                                                                          COMCIDP    8365    A
COMCIDP  *         EXIT   (X6)  =  CONVERTED UPPER 30 BITS OF (X1) -- DPC RESULT    COMCIDP    8366    A
COMCIDP  *                (X7)  =  CONVERTED LOWER 30 BITS OF (X1) -- DPC RESULT    COMCIDP    8367    A
COMCIDP  *                (X0)  =  .ZR. IF BAD POS OR BIT COUNT ON ENTRY,           COMCIDP    8368    A
COMCIDP  *                            ELSE .NZ.                                     COMCIDP    8369    A
COMCIDP  *                                                                          COMCIDP    8370    A
COMCIDP  *         USES   X - 0,1,2,3,4,6,7                                         COMCIDP    8371    A
COMCIDP  *                A - NONE                                                  COMCIDP    8372    A
COMCIDP  *                B - 6                                                     COMCIDP    8373    A
COMCIDP  *                                                                          COMCIDP    8374    A
COMCIDP  *         CALLS  NONE                                                      COMCIDP    8375    A
COMCIDP                                                                             COMCIDP    8376    A
COMCIDP                                                                             COMCIDP    8377    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

```
COMCIDP   VFD      SUBR                 ** ENTRY/EXIT **                          COMCIDP     8378     A
COMCIDP            MX3    0                                                      COMCIDP     8379     A
COMCIDP            BX0    X0-X0                                                  COMCIDP     8380     A
COMCIDP            SB6    B5-1                                                   COMCIDP     8381     A
COMCIDP            LT     B4,B0,EXIT. IF POS COUNT IS BAD                        COMCIDP     8382     A
COMCIDP            LT     B5,B0,EXIT. IF BIT COUNT IS BAD                        COMCIDP     8383     A
COMCIDP            MX0    -3                                                     COMCIDP     8384     A
COMCIDP            ZR     B5,VFD2     IF NOT CONVERTING ANY BITS                 COMCIDP     8385     A
COMCIDP            MX3    1                                                      COMCIDP     8386     A
COMCIDP            AX3    X3,B6                                                  COMCIDP     8387     A
COMCIDP            NO                                                            COMCIDP     8388     A
COMCIDP            LX3    X3,B4       (X3) = EXTRACT MASK FOR BITS TO CONVERT    COMCIDP     8389     A
COMCIDP            BX1    X3*X1                                                  COMCIDP     8390     A
COMCIDP                                                                          COMCIDP     8391     A
COMCIDP   VFD2     MX6    0                                                      COMCIDP     8392     A
COMCIDP            BX7    X7-X7                                                  COMCIDP     8393     A
COMCIDP            SB6    60                                                     COMCIDP     8394     A
COMCIDP                                                                          COMCIDP     8395     A
COMCIDP   *        ASSEMBLE APPROPRIATE DIGITS.                                  COMCIDP     8396     A
COMCIDP                                                                          COMCIDP     8397     A
COMCIDP   VFD3     LX1    3                                                      COMCIDP     8398     A
COMCIDP            BX2    -X0*X1                                                 COMCIDP     8399     A
COMCIDP            LX3    3                                                      COMCIDP     8400     A
COMCIDP            BX4    -X0*X3                                                 COMCIDP     8401     A
COMCIDP            SX2    X2+1R0                                                 COMCIDP     8402     A
COMCIDP            SB6    B6-6                                                   COMCIDP     8403     A
COMCIDP            NZ     X4,VFD4     IF ASSEMBLING THESE DIGITS                 COMCIDP     8404     A
COMCIDP            SX2    1R                                                     COMCIDP     8405     A
COMCIDP                                                                          COMCIDP     8406     A
COMCIDP   VFD4     LX2    X2,B6                                                  COMCIDP     8407     A
COMCIDP            BX7    X7+X2                                                  COMCIDP     8408     A
COMCIDP            GT     B6,B0,VFD3  IF PACKING REG (X7) NOT FULL               COMCIDP     8409     A
COMCIDP                                                                          COMCIDP     8410     A
COMCIDP   *        HERE IF PACKING REGISTER (X7) IS FULL.                        COMCIDP     8411     A
COMCIDP                                                                          COMCIDP     8412     A
COMCIDP            SB6    60                                                     COMCIDP     8413     A
COMCIDP            NZ     X6,EXIT.    IF FINISHED ASSEMBLING ENTIRE WORD         COMCIDP     8414     A
COMCIDP            BX6    X7                                                     COMCIDP     8415     A
COMCIDP            MX7    0                                                      COMCIDP     8416     A
COMCIDP            EQ     VFD3                                                   COMCIDP     8417     A
COMCIDP   .IDPOS   ENDIF                                                         CCGA018       40     A
COMCIDP   COMCIDP  SPACE  4,10                                                   COMCIDP     8418     A
COMCIDP            BASE   *                                                      COMCIDP     8419     A
COMCIDP   QUAL$    IF     -DEF,QUAL$                                             COMCIDP     8420     A
COMCIDP            QUAL   *                                                      COMCIDP     8421     A
COMCIDP   IDP=     =      /COMCIDP/IDP=                                          COMCIDP     8422     A
COMCIDP   ROL=     =      /COMCIDP/ROL=                                          COMCIDP     8423     A
COMCIDP                                                                          COMCIDP     8424     A
COMCIDP            IF     DEF,/COMCIDP/.OS,1                                     COMCIDP     8425     A
COMCIDP   .OS      =      /COMCIDP/.OS                                           COMCIDP     8426     A
COMCIDP   IDP=ER   =      /COMCIDP/IDP=ER                                        COMCIDP     8427     A
COMCIDP   IDP=ERR  =      /COMCIDP/IDP=ERR                                       COMCIDP     8428     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
            LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCIDP

COMCIDP   IDP=MN   =      /COMCIDP/IDP=MN                                    COMCIDP     8429     A
COMCIDP   IDP=SVB  =      /COMCIDP/IDP=SVB                                   COMCIDP     8430     A
COMCIDP   IDP=SVA  =      /COMCIDP/IDP=SVA                                   COMCIDP     8431     A
COMCIDP   IDP=SVX  =      /COMCIDP/IDP=SVX                                   COMCIDP     8432     A
COMCIDP                                                                      COMCIDP     8433     A
COMCIDP   REG=     =      /COMCIDP/REG=                                      COMCIDP     8434     A
COMCIDP   SNP=     =      /COMCIDP/SNP=                                      COMCIDP     8435     A
COMCIDP   QUAL$    ENDIF                                                     COMCIDP     8436     A
COMCIDP   COMCIDP  ENDX                                                      COMCIDP     8437     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - COMCIDP

            IDENTIFIER               TOTAL    ACTIVE

            COMCIDP                   8437      8435
            CCGA018                     36        36

            LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCLFM

COMCLFM  *COMDECK COMCLFM                                                    COMCLFM        1     A
COMCLFM           CTEXT  COMCLFM - LOCAL FILE MANAGER PROCESSOR.             COMCLFM        2     A
COMCLFM   LFM     SPACE  4                                                   COMCLFM        3     A
COMCLFM           IF     -DEF,QUAL$,1                                        COMCLFM        4     A
COMCLFM           QUAL   COMCLFM                                             COMCLFM        5     A
COMCLFM           BASE   D                                                   COMCLFM        6     A
COMCLFM  *        COMMENT COPYRIGHT CONTROL DATA CORP. 1970.                 COMCLFM        7          I
         -CCG*94
COMCLFM  *        COMMENT COPYRIGHT CONTROL DATA SYSTEMS, INC. 1994.         CCG*94        10     A
COMCLFM   LFM     SPACE  4                                                   COMCLFM        8     A
COMCLFM  ***      LFM - LOCAL FILE MANAGER PROCESSOR.                        COMCLFM        9     A
COMCLFM  *        G. R. MANSFIELD.  70/10/09.                                COMCLFM       10     A
COMCLFM   LFM     SPACE  4                                                   COMCLFM       11     A
COMCLFM  ***             LFM PROCESSES REQUESTS FOR THE LOCAL FILE MANAGER PP COMCLFM   12     A
COMCLFM  *        PROGRAM *LFM*.                                             COMCLFM       13     A
COMCLFM  *        IF THE SPECIFIED FET IS BUSY, A FILE RECALL IS PERFORMED.  COMCLFM       14     A
COMCLFM  *        IF THE SPECIFIED FET IS BLANK, LFM RETURNS WITH NO ACTION  COMCLFM       15     A
COMCLFM  *        TAKEN.                                                     COMCLFM       16     A
COMCLFM  *                                                                   COMCLFM       17     A
COMCLFM  *        ENTRY  (X2) = ADDRESS OF FET.                              COMCLFM       18     A
COMCLFM  *               (X7) = FUNCTION CODE.                               COMCLFM       19     A
COMCLFM  *                                                                   COMCLFM       20     A
COMCLFM  *        EXIT   (X2) = ADDRESS OF FET.                              COMCLFM       21     A
COMCLFM  *                                                                   COMCLFM       22     A
COMCLFM  *        USES   X - 1, 6, 7.                                        COMCLFM       23     A
COMCLFM  *               B - NONE.                                           COMCLFM       24     A
COMCLFM  *               A - 1.                                              COMCLFM       25     A
COMCLFM  *                                                                   COMCLFM       26     A
COMCLFM  *        CALLS  SYS=.                                               COMCLFM       27     A
COMCLFM                                                                      COMCLFM       28     A
COMCLFM                                                                      COMCLFM       29     A
COMCLFM   LFM2    SX2    X2                                                  COMCLFM       30     A
COMCLFM           RJ     =XSYS=                                              COMCLFM       31     A

            0         1         2         3         4         5         6         7         8
            12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCLFM

COMCLFM                                                                    COMCLFM        32     A
COMCLFM   LFM=     PS                 ENTRY/EXIT                           COMCLFM        33     A
COMCLFM            SA1    X2          CHECK FILE STATUS                    COMCLFM        34     A
COMCLFM            LX1    59                                               COMCLFM        35     A
COMCLFM            NG     X1,LFM1     IF NOT BUSY                          COMCLFM        36     A
COMCLFM            ZR     X1,LFM=     RETURN IF BLANK FET                  COMCLFM        37     A
COMCLFM            RECALL X2                                               COMCLFM        38     A
COMCLFM   LFM1     SA1    X2          SET FET BUSY                         COMCLFM        39     A
COMCLFM            MX6    59                                               COMCLFM        40     A
COMCLFM            BX6    X6*X1                                            COMCLFM        41     A
COMCLFM            SA6    A1                                               COMCLFM        42     A
COMCLFM            SA1    LFMA        SET *LFM* REQUEST                    COMCLFM        43     A
COMCLFM            LX7    24                                               COMCLFM        44     A
COMCLFM            BX1    X1+X7                                            COMCLFM        45     A
COMCLFM            BX6    X1+X2                                            COMCLFM        46     A
COMCLFM            EQ     LFM2                                             COMCLFM        47     A
COMCLFM                                                                    COMCLFM        48     A
COMCLFM   LFMA     CON    0LLFM+1S40                                       COMCLFM        49     A
COMCLFM            SPACE  4                                                COMCLFM        50     A
COMCLFM            BASE   *                                                COMCLFM        51     A
COMCLFM   QUAL$    IF     -DEF,QUAL$                                       COMCLFM        52     A
COMCLFM            QUAL   *                                                COMCLFM        53     A
COMCLFM   LFM=     EQU    /COMCLFM/LFM=                                    COMCLFM        54     A
COMCLFM   QUAL$    ENDIF                                                   COMCLFM        55     A
COMCLFM            ENDX                                                    COMCLFM        56     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - COMCLFM

          IDENTIFIER                TOTAL    ACTIVE

          COMCLFM                      56        55
          CCG*94                        1         1

          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCMCS

COMCMCS  *COMDECK COMCMCS                                                  COMCMCS         1     A
COMCMCS   MCS      CTEXT  MCS - MERGE CODED STRINGS.                       COMCMCS         2     A
COMCMCS   MCS      SPACE  4                                                COMCMCS         3     A
COMCMCS            IF     -DEF,QUAL$,1                                     COMCMCS         4     A
COMCMCS            QUAL   COMCMCS                                          COMCMCS         5     A
COMCMCS   MCS      SPACE  4                                                COMCMCS         6     A
COMCMCS  ***       MCS - MERGE CODED STRINGS.                              COMCMCS         7     A
COMCMCS  *                                                                 COMCMCS         8     A
COMCMCS  *         D.C. DILLON   75/07/01                                  COMCMCS         9     A
COMCMCS  *         D.C. DILLON   77/02/22                                  COMCMCS        10     A
COMCMCS   MCS      SPACE  4                                                COMCMCS        11     A
COMCMCS  **        MCS - MERGE CODED STRINGS.                              COMCMCS        12     A
COMCMCS  *                                                                 COMCMCS        13     A
COMCMCS  *                CONCATENATES NEW CHARACTER STRING IN (X1) WITH OLD COMCMCS       14     A
COMCMCS  *         STRING AT (B7).  STORES RESULT AT (B7) AND, IF NEW STRING COMCMCS       15     A
COMCMCS  *         LENGTH REQUIRES, AT (B7)+1.  BOTH OLD AND NEW STRINGS ARE COMCMCS       16     A
COMCMCS  *         ASSUMED TO BE LEFT JUSTIFIED WITH ZERO FILL.  EITHER MAY BE COMCMCS     17     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCMCS

```
COMCMCS  *         NULL OR FULL.                                                COMCMCS    18    A
COMCMCS  *                                                                      COMCMCS    19    A
COMCMCS  *               ***  LIMITATION -- TRAILING COLONS ARE IDENTICAL TO   COMCMCS    20    A
COMCMCS  *         ZERO FILL, AND WILL BE LOST.                                 COMCMCS    21    A
COMCMCS  *                                                                      COMCMCS    22    A
COMCMCS  *                                                                      COMCMCS    23    A
COMCMCS  *         ENTRY  (X1) = *NEW* STRING.                                  COMCMCS    24    A
COMCMCS  *                (B1) = 1 IF SYMBOL B1=1 IS DEFINED.                   COMCMCS    25    A
COMCMCS  *                (B7) = ADDRESS OF *OLD* STRING.                       COMCMCS    26    A
COMCMCS  *                                                                      COMCMCS    27    A
COMCMCS  *         EXIT   MERGED STRING STORED AT (B7) AND, POSSIBLY, AT (B7)+1. COMCMCS    28    A
COMCMCS  *                (X1) = UNCHANGED.                                     COMCMCS    29    A
COMCMCS  *                (X2) = ORIGINAL *OLD* STRING.                         COMCMCS    30    A
COMCMCS  *                (X6) = *OLD* .AND. UPPER *NEW*.                       COMCMCS    31    A
COMCMCS  *                (X7) = LOWER *NEW*, IF ANY, ELSE = 0.                 COMCMCS    32    A
COMCMCS  *                (B1) = 1 IF SYMBOL B1=1 IS NOT DEFINED.               COMCMCS    33    A
COMCMCS  *                (B7) = ADDRESS OF *OLD*, UPDATED IF LOWER *NEW* .NZ.  COMCMCS    34    A
COMCMCS  *                                                                      COMCMCS    35    A
COMCMCS  *         USES   X - 2, 3, 6, 7                                        COMCMCS    36    A
COMCMCS  *                A - 2, 3, 6, 7                                        COMCMCS    37    A
COMCMCS  *                B - 1, 2, 7                                           COMCMCS    38    A
COMCMCS  *                                                                      COMCMCS    39    A
COMCMCS  *         CALLS  NONE                                                  COMCMCS    40    A
COMCMCS                                                                         COMCMCS    41    A
COMCMCS                                                                         COMCMCS    42    A
COMCMCS   MCS      SUBR                ** ENTRY/EXIT **                         COMCMCS    43    A
COMCMCS                                                                         COMCMCS    44    A
COMCMCS            IF     -DEF,B1=1,1                                           COMCMCS    45    A
COMCMCS            SB1    1                                                     COMCMCS    46    A
COMCMCS                                                                         COMCMCS    47    A
COMCMCS            SA2    B7          (X2) = *OLD* STRING                       COMCMCS    48    A
COMCMCS            MX7    -1                                                    COMCMCS    49    A
COMCMCS            SA3    MCSA                                                  COMCMCS    50    A
COMCMCS            IX6    X2+X7       BORROW RIPPLES LEFT TO 1ST NON-ZERO CHAR  COMCMCS    51    A
COMCMCS            BX7    -X2*X6      EXTRACT BORROWS                           COMCMCS    52    A
COMCMCS            SB2    55D                                                   COMCMCS    53    A
COMCMCS            BX7    X3*X7       EXTRACT NULL BYTES = 40                   COMCMCS    54    A
COMCMCS            LX3    X7,B2       EACH NULL BYTE     = 01                   COMCMCS    55    A
COMCMCS            IX6    X7-X3                          = 37                   COMCMCS    56    A
COMCMCS            BX7    X6+X7       EACH NULL BYTE     = 77 ... EXTRACTION MASK COMCMCS    57    A
COMCMCS            CX6    X7                                                    COMCMCS    58    A
COMCMCS            SB2    X6                                                    COMCMCS    59    A
COMCMCS            LX3    X1,B2       ALIGN *NEW* WITH NULL PART OF *OLD*       COMCMCS    60    A
COMCMCS            BX6    X7*X3                                                 COMCMCS    61    A
COMCMCS            BX7    -X7*X3      (X7) = LOWER *NEW*                        COMCMCS    62    A
COMCMCS            BX6    X2+X6       (X6) = *OLD* .OR. UPPER *NEW*             COMCMCS    63    A
COMCMCS            SA6    B7+0                                                  COMCMCS    64    A
COMCMCS            NZ     X7,MCS2     IF LOWER *NEW* NOT EMPTY                  COMCMCS    65    A
COMCMCS            PL     X7,EXIT.                                              COMCMCS    66    A
COMCMCS   MCS2     SA7    B7+B1                                                 COMCMCS    67    A
COMCMCS            SB7    B7+B1       (B7) = UPDATED STRING ADDRESS             COMCMCS    68    A
COMCMCS            EQ     EXIT.                                                 COMCMCS    69    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCMCS

COMCMCS                                                                                COMCMCS       70     A
COMCMCS                                                                                COMCMCS       71     A
COMCMCS   MCSA     DATA   40404040404040404040B   NULL CHARACTER BYTE MASK             COMCMCS       72     A
COMCMCS   MCS      SPACE  4                                                             COMCMCS       73     A
COMCMCS   QUAL$    IF     -DEF,QUAL$                                                    COMCMCS       74     A
COMCMCS            QUAL   *                                                             COMCMCS       75     A
COMCMCS   MCS      =      /COMCMCS/MCS                                                  COMCMCS       76     A
COMCMCS   QUAL$    ENDIF                                                                COMCMCS       77     A
COMCMCS            ENDX                                                                 COMCMCS       78     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - COMCMCS

          IDENTIFIER                TOTAL    ACTIVE

          COMCMCS                      78        78

          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCSBM

COMCSBM  *COMDECK  COMCSBM            SET BLOCK OF MEMORY TO GIVEN VALUE                COMCSBM        1     A
COMCSBM   SBM      CTEXT  SET BLOCK OF MEMORY TO A GIVEN VALUE                          COMCSBM        2     A
COMCSBM   SBM      SPACE  4,8                                                           COMCSBM        3     A
COMCSBM            IF     -DEF,QUAL$,1                                                  COMCSBM        4     A
COMCSBM            QUAL   COMCSBM                                                       COMCSBM        5     A
COMCSBM            BASE   D                                                             COMCSBM        6     A
COMCSBM  *         COMMENT COPYRIGHT CONTROL DATA CORP. 1976.                           COMCSBM        7           I
          -CCG*94
COMCSBM  *         COMMENT COPYRIGHT CONTROL DATA SYSTEMS, INC. 1994.                   CCG*94        11     A
COMCSBM   SBM      SPACE  4,8                                                           COMCSBM        8     A
COMCSBM  **        SBM - SET BLOCK OF MEMORY TO A GIVEN VALUE.                          COMCSBM        9     A
COMCSBM  *                                                                              COMCSBM       10     A
COMCSBM  *         ENTRY  (X1) = BLOCK LENGTH                                           COMCSBM       11     A
COMCSBM  *                (X6) = VALUE TO SET                                           COMCSBM       12     A
COMCSBM  *                (A6) = BLOCK FWA (WORD 1 ALREADY STORED)                      COMCSBM       13     A
COMCSBM  *                                                                              COMCSBM       14     A
COMCSBM  *         EXIT   (X6) = UNCHANGED                                              COMCSBM       15     A
COMCSBM  *                (X7) = (X6)                                                   COMCSBM       16     A
COMCSBM  *                (A6) = BLOCK LWA                                              COMCSBM       17     A
COMCSBM  *                (B1) = 1                                                      COMCSBM       18     A
COMCSBM  *                                                                              COMCSBM       19     A
COMCSBM  *         USES   X - 1, 2, 7                                                   COMCSBM       20     A
COMCSBM  *                A - 6, 7                                                      COMCSBM       21     A
COMCSBM  *                B - 1                                                         COMCSBM       22     A
COMCSBM  *                                                                              COMCSBM       23     A
COMCSBM  *         CALLS  NONE                                                          COMCSBM       24     A
COMCSBM                                                                                 COMCSBM       25     A
COMCSBM                                                                                 COMCSBM       26     A
COMCSBM   SBM2     SA7    A6+1                                                          COMCSBM       27     A
COMCSBM            IX1    X1-X2                                                         COMCSBM       28     A
COMCSBM            SA6    A7+B1                                                         COMCSBM       29     A
COMCSBM            PL     X1,SBM2     IF ALL OF BLOCK NOT SET                           COMCSBM       30     A
COMCSBM                                                                                 COMCSBM       31     A
COMCSBM   SBM=     SUBR   0           ENTRY/EXIT...                                     COMCSBM       32     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCSBM

COMCSBM            SX2    1                                                    COMCSBM      33     A
COMCSBM            NO                                                          COMCSBM      34     A
COMCSBM            BX7    -X1*X2      EXTRACT COMPLEMENT OF LSB                COMCSBM      35     A
COMCSBM            SB1    X7          (B1) = 1 IF LENGTH EVEN, = 0 IF ODD      COMCSBM      36     A
COMCSBM            SX1    X1-3        DECREMENT LENGTH                         COMCSBM      37     A
COMCSBM            SA6    A6+B1       SET 2ND WORD IF LENGTH WAS EVEN          COMCSBM      38     A
COMCSBM            LX2    1           (X2) = 2                                 COMCSBM      39     A
COMCSBM            SB1    1                                                    COMCSBM      40     A
COMCSBM            BX7    X6                                                   COMCSBM      41     A
COMCSBM            PL     X1,SBM2     IF ALL OF BLOCK NOT SET                  COMCSBM      42     A
COMCSBM            EQ     EXIT.       EXIT...                                  COMCSBM      43     A
COMCSBM   SBM      SPACE  4,8                                                  COMCSBM      44     A
COMCSBM            BASE   *                                                    COMCSBM      45     A
COMCSBM   QUAL$    IF     -DEF,QUAL$                                           COMCSBM      46     A
COMCSBM            QUAL   *                                                    COMCSBM      47     A
COMCSBM   SBM=     EQU    /COMCSBM/SBM=                                        COMCSBM      48     A
COMCSBM   QUAL$    ENDIF                                                       COMCSBM      49     A
COMCSBM            ENDX                                                        COMCSBM      50     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - COMCSBM

          IDENTIFIER                TOTAL    ACTIVE

          COMCSBM                      50        49
          CCG*94                        1         1

          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCTOK

COMCTOK  *COMDECK COMCTOK                                                      COMCTOK       1     A
COMCTOK   TOK      CTEXT  COMCTOK 2.0 - TOKEN GENERATOR.                       COMCTOK       2     A
COMCTOK   TOK      SPACE  4,10                                                 COMCTOK       3     A
COMCTOK            IF     -DEF,QUAL$,1                                         COMCTOK       4     A
COMCTOK            QUAL   COMCTOK                                              COMCTOK       5     A
COMCTOK            BASE   D                                                    COMCTOK       6     A
COMCTOK   TOK      SPACE  4,10                                                 COMCTOK       7     A
COMCTOK  ***       TOK - TOKEN GENERATOR.                                      COMCTOK       8     A
COMCTOK  *         P CLARE  77/01/20                                           COMCTOK       9     A
COMCTOK  *         P CLARE  78/04/27  VERSION 2.0                              COMCTOK      10     A
COMCTOK   TOK      SPACE  4,10                                                 COMCTOK      11     A
COMCTOK  ***       TOK - TOKEN GENERATOR.                                      COMCTOK      12     A
COMCTOK  *                                                                     COMCTOK      13     A
COMCTOK  *                                                                     COMCTOK      14     A
COMCTOK  *         *COMCTOK* TRANSLATES AN INPUT SOURCE LINE FROM ITS PACKED   COMCTOK      15     A
COMCTOK  *         (10 CHARACTER PER WORD) FORM TO ITS ENTOKENED FORM.  THIS   COMCTOK      16     A
COMCTOK  *         *ENTOKENED FORM* IS DESCRIBED BY A HUMAN USING THE *TOGEL*  COMCTOK      17     A
COMCTOK  *         MACROS (DEFINED IN *COMATOK*).  THE PROGRAMMER ENCODES      COMCTOK      18     A
COMCTOK  *         THE *TOGEL* MACROS IN SUCH A FASHION AS TO DESCRIBE THE     COMCTOK      19     A
COMCTOK  *         PROCESS BY WHICH *TOK* CAN ENTOKEN ANY INPUT SOURCE LINE    COMCTOK      20     A
COMCTOK  *         THAT THE PROGRAMMER IS INTERESTED IN.                       COMCTOK      21     A
COMCTOK  *                                                                     COMCTOK      22     A
COMCTOK  *         THE *TOGEL* MACROS GENERATE A *TOGEL* OBJECT MODULE,        COMCTOK      23     A
COMCTOK  *         CALLED *TOM*, WHICH THE PROGRAMMER THEN FEEDS TO *TOK*.     COMCTOK      24     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCTOK

```
COMCTOK  *        *TOK* WILL EXECUTE/INTERPRET A *TOM*, PERFORMING THE          COMCTOK    25    A
COMCTOK  *        SPECIFIED ENTOKENING TASKS UPON THE INPUT SOURCE LINE.        COMCTOK    26    A
COMCTOK  *                                                                      COMCTOK    27    A
COMCTOK  *        *TOK* CAN INTERPRET THE FOLLOWING *TOGEL* BINARY              COMCTOK    28    A
COMCTOK  *        INSTRUCTIONS --                                               COMCTOK    29    A
COMCTOK  *                                                                      COMCTOK    30    A
COMCTOK  *              GROUP (A..Z),TOT,SQZ                                    COMCTOK    31    A
COMCTOK  *              GROUP (A..Z),TOT,NSQZ                                   COMCTOK    32    A
COMCTOK  *              IFT (A..Z)                                              COMCTOK    33    A
COMCTOK  *                THEN                                                  COMCTOK    34    A
COMCTOK  *                ELST                                                  COMCTOK    35    A
COMCTOK  *                ENDT                                                  COMCTOK    36    A
COMCTOK  *              GOTO XX                                                 COMCTOK    37    A
COMCTOK  *              CALT XX                                                 COMCTOK    38    A
COMCTOK  *              CASEOF (A..Z)                                           COMCTOK    39    A
COMCTOK  *                TOKEN                                                 COMCTOK    40    A
COMCTOK  *              ELSC                                                    COMCTOK    41    A
COMCTOK  *                TOKEN                                                 COMCTOK    42    A
COMCTOK  *              ENDC                                                    COMCTOK    43    A
COMCTOK  *              GENTOK TOT                                              COMCTOK    44    A
COMCTOK  *              XEQ XXX                                                 COMCTOK    45    A
COMCTOK  *              PROC XXX                                                COMCTOK    46    A
COMCTOK  *                                                                      COMCTOK    47    A
COMCTOK  *        THE TOKEN GENERATION PROCESS, THEN, IS EXTREMELY TABLE DRIVEN.COMCTOK    48    A
COMCTOK  *        *TOK* HAS VERY LITTLE INTERNAL LOGIC OF ITS OWN.  IT DOES     COMCTOK    49    A
COMCTOK  *        EXACTLY WHAT THE *TOM* THAT IS DRIVING IT TELLS IT TO DO.     COMCTOK    50    A
COMCTOK  *                                                                      COMCTOK    51    A
COMCTOK  *        EACH PROCESSING NODE WITHIN *TOK* EXECUTES/INTERPRETS A       COMCTOK    52    A
COMCTOK  *        SINGLE *TOGEL* BINARY INSTRUCTION FROM THE *TOM*.  THESE      COMCTOK    53    A
COMCTOK  *        *TOGEL* BINARY INSTRUCTION INTERPRETERS ARE CALLED            COMCTOK    54    A
COMCTOK  *        *TOK* FUNCTIONAL UNITS, *TOFU*S FOR SHORT.                    COMCTOK    55    A
COMCTOK  *                                                                      COMCTOK    56    A
COMCTOK  *        THE ENTIRE SUBJECT OF TOKEN GENERATION AND HOW TO UNDERSTAND  COMCTOK    57    A
COMCTOK  *        AND USE *TOGEL* AND *TOK* IS DEALT WITH MUCH MORE THOROUGHLY  COMCTOK    58    A
COMCTOK  *        IN THE FTN 5 IMS IN THE CHAPTER FOR THE DECK *LEX*.           COMCTOK    59    A
COMCTOK  *        I STRONGLY ADVISE READING IT BEFORE TRYING TO MESS IN THIS    COMCTOK    60    A
COMCTOK  *        CODE.  IT WILL SAVE YOU A LOT OF TIME...                      COMCTOK    61    A
COMCTOK  *                                                                      COMCTOK    62    A
COMCTOK  ***********************************************************************COMCTOK    63    A
COMCTOK  *                                                                      COMCTOK    64    A
COMCTOK  *        USING *COMCTOK*                                               COMCTOK    65    A
COMCTOK  *                                                                      COMCTOK    66    A
COMCTOK  *        USING *COMCTOK* INVOLVES DOING THE FOLLOWING --               COMCTOK    67    A
COMCTOK  *                                                                      COMCTOK    68    A
COMCTOK  *          1. CONFIGURING *COMSTOK* (COMMON TOKEN GENERATOR INTERFACE  COMCTOK    69    A
COMCTOK  *             TEXT), *COMATOK* (COMMON TOKEN GENERATOR TOGEL MACROS),  COMCTOK    70    A
COMCTOK  *             AND *COMCTOK* (COMMON TOKEN GENERATOR).                  COMCTOK    71    A
COMCTOK  *                                                                      COMCTOK    72    A
COMCTOK  *             THIS ALSO INVOLVES THE SETTING UP OF SOME ASSEMBLY-TIME  COMCTOK    73    A
COMCTOK  *             DEPENDENCIES.  SEE BELOW.                                COMCTOK    74    A
COMCTOK  *                                                                      COMCTOK    75    A
COMCTOK  *          2. LEARNING ENOUGH ABOUT *TOGEL* (TOKEN GENERATION LANGUAGE)COMCTOK    76    A
```

```
         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCTOK

```
COMCTOK  *              TO WRITE A *TOGEL* PROGRAM THAT WILL DIRECT *TOK* AS TO          COMCTOK     77    A
COMCTOK  *              HOW YOU WISH HIM TO ENTOKEN.  I SUGGEST A READING OF THE         COMCTOK     78    A
COMCTOK  *              FTN 5 IMS FOR THE DECK *LEX*, AS THERE IS A THOROUGH             COMCTOK     79    A
COMCTOK  *              DESCRIPTION OF *TOGEL* PROGRAMMING THERE.                        COMCTOK     80    A
COMCTOK  *                                                                               COMCTOK     81    A
COMCTOK  *           3. WRITING YOUR *TOGEL* PROGRAM USING THE MACROS CONTAINED          COMCTOK     82    A
COMCTOK  *              IN *COMATOK* (COMMON TOKEN GENERATOR *TOGEL* MACROS).            COMCTOK     83    A
COMCTOK  *                                                                               COMCTOK     84    A
COMCTOK  *           4. SETTING UP *TOKCOM* (THE USER/TOK COMMUNICATIONS AREA).          COMCTOK     85    A
COMCTOK  *              *TOKCOM* IS CONCEPTUALLY SOMETHING LIKE A FILE *FET*             COMCTOK     86    A
COMCTOK  *              IN THAT IT IS A PARAMETER LIST THAT IS USED BOTH FOR             COMCTOK     87    A
COMCTOK  *              INPUT-TO-TOK AND OUTPUT-FROM-TOK INFORMATION.                    COMCTOK     88    A
COMCTOK  *                                                                               COMCTOK     89    A
COMCTOK  *              IT SHOULD CONTAIN, AMONG OTHER THINGS, THE ADDR THAT YOU         COMCTOK     90    A
COMCTOK  *              WANT *TOK* TO ENTOKEN FROM, AND THE ADDR THAT YOU WISH           COMCTOK     91    A
COMCTOK  *              *TOK* TO ENTOKEN TO.  SEE NEXT SECTION FOR MORE                  COMCTOK     92    A
COMCTOK  *              INFORMATION ON *TOKCOM*.                                         COMCTOK     93    A
COMCTOK  *                                                                               COMCTOK     94    A
COMCTOK  *           5. CALLING *TOK* (COMMON TOKEN GENERATOR) VIA AN *RJ* WITH          COMCTOK     95    A
COMCTOK  *              (B2) SET TO THE FWA OF YOUR *TOKCOM*.                            COMCTOK     96    A
COMCTOK  *                                                                               COMCTOK     97    A
COMCTOK  *              NOTE THAT THIS CALLING SEQUENCE ALLOWS *TOK* TO BE USED          COMCTOK     98    A
COMCTOK  *              WITH MULTIPLE *TOKCOM*S.                                         COMCTOK     99    A
COMCTOK  *                                                                               COMCTOK    100    A
COMCTOK  ************************************************************************       COMCTOK    101    A
COMCTOK  *                                                                               COMCTOK    102    A
COMCTOK  *         CONFIGURING *COMCTOK*                                                 COMCTOK    103    A
COMCTOK  *                                                                               COMCTOK    104    A
COMCTOK  *         *COMCTOK* REQUIRES THE FOLLOWING COMDECKS --                          COMCTOK    105    A
COMCTOK  *                                                                               COMCTOK    106    A
COMCTOK  *                COMACPU - GENERAL CPU MACROS (IN *CPUTEXT*).                   COMCTOK    107    A
COMCTOK  *                                                                               COMCTOK    108    A
COMCTOK  *                COMADEF - STRUCTURED FIELD DECLARATION MACROS.                 COMCTOK    109    A
COMCTOK  *                                                                               COMCTOK    110    A
COMCTOK  *                COMAREG - REPLACEMENT FOR R= PSEUDO INSTRUCTION                COMCTOK    111    A
COMCTOK  *                          (IN *CPUTEXT*).                                      COMCTOK    112    A
COMCTOK  *                                                                               COMCTOK    113    A
COMCTOK  *                COMATOK - COMMON TOKEN GENERATOR TOGEL MACROS.                 COMCTOK    114    A
COMCTOK  *                                                                               COMCTOK    115    A
COMCTOK  *                COMCBUB - BURST/BUILD CHARACTERS WITH BLANK SQUEEZE.           COMCTOK    116    A
COMCTOK  *                                                                               COMCTOK    117    A
COMCTOK  *                COMCBUN - BURST/BUILD CHARACTERS WITH NO BLANK SQUEEZE.        COMCTOK    118    A
COMCTOK  *                                                                               COMCTOK    119    A
COMCTOK  *                COMSTOK - COMMON TOKEN GENERATOR INTERFACE TEXT.               COMCTOK    120    A
COMCTOK  *                                                                               COMCTOK    121    A
COMCTOK  *         IN ADDITION, IF YOU DECIDE TO INSTALL *COMCTOK* IN *TEST* MODE        COMCTOK    122    A
COMCTOK  *         (SEE BELOW), THEN *COMDTOK* (COMMON TOKEN GENERATOR IDP               COMCTOK    123    A
COMCTOK  *         UTILITY) MUST BE PROVIDED (ALONG WITH A PROPERLY CONFIGURED           COMCTOK    124    A
COMCTOK  *         *COMCIDP*, OF COURSE).                                                COMCTOK    125    A
COMCTOK  *                                                                               COMCTOK    126    A
COMCTOK  *         IN THE FOLLOWING DISCUSSION, *ASSEMBLY-TIME DEPENDENCIES*             COMCTOK    127    A
COMCTOK  *         ARE DEFINED TO BE THOSE INTERFACES THAT THE *COMCTOK*                 COMCTOK    128    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCTOK

```
COMCTOK  *         INSTALLER **MUST** PROVIDE, WHILE *ASSEMBLY-TIME OPTIONS* ARE      COMCTOK    129    A
COMCTOK  *         THOSE INTERFACES THAT ARE OPTIONAL.                                COMCTOK    130    A
COMCTOK  *                                                                            COMCTOK    131    A
COMCTOK  *-------------------------------------------------------------------------   COMCTOK    132    A
COMCTOK  *                                                                            COMCTOK    133    A
COMCTOK  *         ASSEMBLY-TIME DEPENDENCIES --                                      COMCTOK    134    A
COMCTOK  *                                                                            COMCTOK    135    A
COMCTOK  *-------------------------------------------------------------------------   COMCTOK    136    A
COMCTOK  *                                                                            COMCTOK    137    A
COMCTOK  *         BB./BN. STRUCTURE DEFINITIONS.                                     COMCTOK    138    A
COMCTOK  *                                                                            COMCTOK    139    A
COMCTOK  *         *BB.* AND *BN.* ARE THE STRUCTURE DEFINITIONS FOR *COMCBUB*        COMCTOK    140    A
COMCTOK  *         AND *COMCBUN*, RESPECTIVELY.  THESE STRUCTURES DESCRIBE/DEFINE     COMCTOK    141    A
COMCTOK  *         WHAT A *GROUP* GENERATED VIA *GROUP,,SQZ/COMCBUB* AND/OR           COMCTOK    142    A
COMCTOK  *         *GROUP,,NSQZ/COMCBUN* LOOKS LIKE.                                  COMCTOK    143    A
COMCTOK  *                                                                            COMCTOK    144    A
COMCTOK  *         FIELDS WITHIN *BB./BN.* STRUCTURES ARE AS FOLLOWS --               COMCTOK    145    A
COMCTOK  *                                                                            COMCTOK    146    A
COMCTOK  *               TOC = TOKEN CHARACTERS.  I.E. *COMCBUB/COMCBUN* WILL         COMCTOK    147    A
COMCTOK  *                     BURST/BUILD CHARACTERS INTO THIS FIELD.                COMCTOK    148    A
COMCTOK  *                                                                            COMCTOK    149    A
COMCTOK  *               TOT = TOKEN TYPE.  I.E. *COMCTOK* WILL FILL THIS             COMCTOK    150    A
COMCTOK  *                     FIELD WITH THE SPECIFIED TOKEN TYPE.                   COMCTOK    151    A
COMCTOK  *                                                                            COMCTOK    152    A
COMCTOK  *                     NOTE THAT *TOT* FIELD CAN BE OF NULL LENGTH            COMCTOK    153    A
COMCTOK  *                     IF DESIRED.  I.E. --                                   COMCTOK    154    A
COMCTOK  *                                                                            COMCTOK    155    A
COMCTOK  *                             DESCRIBE BN.                                   COMCTOK    156    A
COMCTOK  *                         TOC  DEFINE   60                                   COMCTOK    157    A
COMCTOK  *                         TOT  DEFINE   0                                    COMCTOK    158    A
COMCTOK  *                                                                            COMCTOK    159    A
COMCTOK  *                     THIS CAN BE USEFUL FOR PACKING UP CHARACTER            COMCTOK    160    A
COMCTOK  *                     STRINGS THAT DO NOT HAVE A TOKEN TYPE ASSOCIATED       COMCTOK    161    A
COMCTOK  *                     WITH THEM.                                             COMCTOK    162    A
COMCTOK  *                                                                            COMCTOK    163    A
COMCTOK  *-------------------------------------------------------------------------   COMCTOK    164    A
COMCTOK  *                                                                            COMCTOK    165    A
COMCTOK  *         "O."XXX - TOKEN (OPERATOR/OPERAND) TYPES.                          COMCTOK    166    A
COMCTOK  *                                                                            COMCTOK    167    A
COMCTOK  *         THE *COMCTOK* INSTALLER MUST DEFINE ALL POSSIBLE TOKEN TYPES       COMCTOK    168    A
COMCTOK  *         USED IN THE TOGEL PROGRAM, WITH THE EXCEPTION OF "O."BOS           COMCTOK    169    A
COMCTOK  *         (BEGINNING OF STMT) AND "O."EOS (END OF STMT) TOKENS.              COMCTOK    170    A
COMCTOK  *         SEE "O."BOS AND "O."EOS IN *ASSEMBLY-TIME OPTIONS* BELOW.          COMCTOK    171    A
COMCTOK  *                                                                            COMCTOK    172    A
COMCTOK  *         TOKEN TYPES ARE COMPASS SYMBOLS THAT HAVE THE FORM --              COMCTOK    173    A
COMCTOK  *                                                                            COMCTOK    174    A
COMCTOK  *               "O."XXX                                                      COMCTOK    175    A
COMCTOK  *                                                                            COMCTOK    176    A
COMCTOK  *         WHERE  "O." = TOKEN TYPE SYMBOL PREFIX MICRO.  SEE "O."            COMCTOK    177    A
COMCTOK  *                      BELOW IN *ASSEMBLY-TIME OPTIONS*.                     COMCTOK    178    A
COMCTOK  *                                                                            COMCTOK    179    A
COMCTOK  *               XXX  = A UNIQUE TOKEN TYPE, AS USED IN TOGEL MACRO           COMCTOK    180    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCTOK

```
COMCTOK  *                      REFERENCES.                                          COMCTOK     181     A
COMCTOK  *                                                                           COMCTOK     182     A
COMCTOK  *                      FOR EXAMPLE --                                       COMCTOK     183     A
COMCTOK  *                                                                           COMCTOK     184     A
COMCTOK  *                         GROUP (0..9),XXX,SQZ                              COMCTOK     185     A
COMCTOK  *                                                                           COMCTOK     186     A
COMCTOK  *                      WILL GROUP THE CHARACTERS 0 THRU 9 INCLUSIVE         COMCTOK     187     A
COMCTOK  *                      INTO THE TOKEN *"O."XXX*.                            COMCTOK     188     A
COMCTOK  *                                                                           COMCTOK     189     A
COMCTOK  *                                                                           COMCTOK     190     A
COMCTOK  *-------------------------------------------------------------------------- COMCTOK     191     A
COMCTOK  *                                                                           COMCTOK     192     A
COMCTOK  *         ASSEMBLY-TIME OPTIONS --                                          COMCTOK     193     A
COMCTOK  *                                                                           COMCTOK     194     A
COMCTOK  *         NOTE THAT FOR THE INSTALLER'S OPTIONS TO OVERRIDE THE DEFAULTS    COMCTOK     195     A
COMCTOK  *         SET UP IN *COMSTOK* (COMMON TOKEN GENERATOR INTERFACE TEXT),      COMCTOK     196     A
COMCTOK  *         THAT THE OVERRIDING OPTIONS MUST **PRECEDE** *COMSTOK*.           COMCTOK     197     A
COMCTOK  *                                                                           COMCTOK     198     A
COMCTOK  *-------------------------------------------------------------------------- COMCTOK     199     A
COMCTOK  *                                                                           COMCTOK     200     A
COMCTOK  *         BICH - BIAS CHARACTER.                                            COMCTOK     201     A
COMCTOK  *                                                                           COMCTOK     202     A
COMCTOK  *         *BICH* SPECIFIES THE CHARACTER THAT IS TO HAVE BINARY VALUE       COMCTOK     203     A
COMCTOK  *         00B (I.E. BIT 59 IN CHARACTER SHIFT MASK) IN *COMCTOK/COMCBUB/    COMCTOK     204     A
COMCTOK  *         COMCBUN*S ROTATED CHARACTER SET.  SEE *COMCBUB* PREAMBLE FOR      COMCTOK     205     A
COMCTOK  *         A DESCRIPTION OF *BICH*.                                          COMCTOK     206     A
COMCTOK  *                                                                           COMCTOK     207     A
COMCTOK  *         DEFAULT:  BICH  =  1RM                                            COMCTOK     208     A
COMCTOK  *                                                                           COMCTOK     209     A
COMCTOK  *-------------------------------------------------------------------------- COMCTOK     210     A
COMCTOK  *                                                                           COMCTOK     211     A
COMCTOK  *         "BLOWUP" - CATASTROPHIC ERROR ADDRESS MICRO.                      COMCTOK     212     A
COMCTOK  *                                                                           COMCTOK     213     A
COMCTOK  *         *COMCTOK* WILL JUMP TO "BLOWUP" IF IT DETECTS AN UNRECOVERABLE    COMCTOK     214     A
COMCTOK  *         ERROR.                                                            COMCTOK     215     A
COMCTOK  *                                                                           COMCTOK     216     A
COMCTOK  *         DEFAULT:  BLOWUP  MICRO  1,,/*O+4S15/                             COMCTOK     217     A
COMCTOK  *                                                                           COMCTOK     218     A
COMCTOK  *-------------------------------------------------------------------------- COMCTOK     219     A
COMCTOK  *                                                                           COMCTOK     220     A
COMCTOK  *         CHAR - CHARACTER WIDTH.                                           COMCTOK     221     A
COMCTOK  *                                                                           COMCTOK     222     A
COMCTOK  *         *CHAR* SPECIFIES THE WIDTH (IN BITS) OF A SINGLE CHARACTER.       COMCTOK     223     A
COMCTOK  *         THIS OPTION IS INTENDED FOR DPC/ASCII CHARACTER SETS, BUT         COMCTOK     224     A
COMCTOK  *         IS USELESS BECAUSE CURRENTLY *COMCTOK* DOES NOT SUPPORT           COMCTOK     225     A
COMCTOK  *         ASCII.                                                            COMCTOK     226     A
COMCTOK  *                                                                           COMCTOK     227     A
COMCTOK  *         DEFAULT:  CHAR  =  6  (I.E. DPC)                                  COMCTOK     228     A
COMCTOK  *                                                                           COMCTOK     229     A
COMCTOK  *-------------------------------------------------------------------------- COMCTOK     230     A
COMCTOK  *                                                                           COMCTOK     231     A
COMCTOK  *         "O." - TOKEN (OPERATOR/OPERAND) TYPE PREFIX MICRO.                COMCTOK     232     A

         0        1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCTOK

```
COMCTOK  *                                                                  COMCTOK    233    A
COMCTOK  *         *COMCTOK* CONSTRUCTS ALL TOKEN TYPE SYMBOLS SPECIFIED ON COMCTOK    234    A
COMCTOK  *         *TOGEL* INSTRUCTION/MACRO REFERENCES BY PREFIXING THE    COMCTOK    235    A
COMCTOK  *         SPECIFIED CHARACTERS WITH "O.".                          COMCTOK    236    A
COMCTOK  *                                                                  COMCTOK    237    A
COMCTOK  *         FOR EXAMPLE --                                           COMCTOK    238    A
COMCTOK  *                                                                  COMCTOK    239    A
COMCTOK  *                GROUP (A..Z),VAR,SQZ                              COMCTOK    240    A
COMCTOK  *                                                                  COMCTOK    241    A
COMCTOK  *         GENERATES A REFERENCE TO THE TOKEN TYPE *VAR*, WHICH     COMCTOK    242    A
COMCTOK  *         IS REALLY A REFERENCE TO THE COMPASS SYMBOL *"O."VAR*.   COMCTOK    243    A
COMCTOK  *                                                                  COMCTOK    244    A
COMCTOK  *         DEFAULT:  O.  MICRO 1,,/O./                              COMCTOK    245    A
COMCTOK  *                                                                  COMCTOK    246    A
COMCTOK  *-----------------------------------------------------------------  COMCTOK    247    A
COMCTOK  *                                                                  COMCTOK    248    A
COMCTOK  *         "O."BOS - BEGINNING OF STATEMENT TOKEN TYPE.             COMCTOK    249    A
COMCTOK  *                                                                  COMCTOK    250    A
COMCTOK  *         *COMCTOK* GENERATES AN *"O."BOS* TOKEN AS THE 1ST TOKEN IN COMCTOK  251    A
COMCTOK  *         THE STATEMENT, IF NOT IN *RESTART* MODE.  SEE *TC.RSF/TC.REST* COMCTOK  252    A
COMCTOK  *         IN *TOKCOM* BELOW.                                       COMCTOK    253    A
COMCTOK  *                                                                  COMCTOK    254    A
COMCTOK  *         DEFAULT:  "O."BOS  =  0                                  COMCTOK    255    A
COMCTOK  *                                                                  COMCTOK    256    A
COMCTOK  *-----------------------------------------------------------------  COMCTOK    257    A
COMCTOK  *                                                                  COMCTOK    258    A
COMCTOK  *         "O."EOS - END OF STATEMENT TOKEN TYPE.                   COMCTOK    259    A
COMCTOK  *                                                                  COMCTOK    260    A
COMCTOK  *         *COMCTOK* GENERATES AN *"O."EOS* TOKEN, BEFORE EXITING, TO COMCTOK  261    A
COMCTOK  *         MARK THE END OF THE STATEMENT IN THE TOKEN BUFFER.       COMCTOK    262    A
COMCTOK  *                                                                  COMCTOK    263    A
COMCTOK  *         DEFAULT:  "O."EOS  =  0                                  COMCTOK    264    A
COMCTOK  *                                                                  COMCTOK    265    A
COMCTOK  *         NOTE THAT THE DEFAULTS FOR "O."BOS AND "O."EOS MEAN THAT A COMCTOK  266    A
COMCTOK  *         STATEMENT IN THE TOKEN BUFFER WILL BE DELIMITED BY 2 FULL ZERO COMCTOK  267    A
COMCTOK  *         WORDS.                                                   COMCTOK    268    A
COMCTOK  *                                                                  COMCTOK    269    A
COMCTOK  *-----------------------------------------------------------------  COMCTOK    270    A
COMCTOK  *                                                                  COMCTOK    271    A
COMCTOK  *         "O=" - *TOKEN* *TOGEL* INSTRUCTION LABEL PREFIX MICRO.   COMCTOK    272    A
COMCTOK  *                                                                  COMCTOK    273    A
COMCTOK  *         IF THE *CASEOF* TOGEL INSTRUCTION MACRO IS USED, THEN TOKENS COMCTOK  274    A
COMCTOK  *         THAT ARE TO BE GENERATED VIA THIS MECHANISM WILL HAVE A  COMCTOK    275    A
COMCTOK  *         COMPASS LABEL THAT IS CONSTRUCTED BY PREFIXING "O=" TO THE COMCTOK  276    A
COMCTOK  *         TOKEN TYPE.                                              COMCTOK    277    A
COMCTOK  *                                                                  COMCTOK    278    A
COMCTOK  *         FOR EXAMPLE --                                           COMCTOK    279    A
COMCTOK  *                                                                  COMCTOK    280    A
COMCTOK  *                CASEOF (+..*)                                     COMCTOK    281    A
COMCTOK  *                  TOKEN PLUS  .... HAS LABEL *"O="PLUS*           COMCTOK    282    A
COMCTOK  *                  TOKEN MIN   .... HAS LABEL *"O="MIN*            COMCTOK    283    A
COMCTOK  *                  TOKEN STAR  .... HAS LABEL *"O="STAR*           COMCTOK    284    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCTOK

```
COMCTOK  *                ENDC                                                  COMCTOK    285    A
COMCTOK  *                                                                      COMCTOK    286    A
COMCTOK  *         DEFAULT:  O=  MICRO  1,,/O=/                                 COMCTOK    287    A
COMCTOK  *                                                                      COMCTOK    288    A
COMCTOK  *         THIS OPTION IS INTENDED PRIMARILY TO AVOID SYMBOL CONFLICTS  COMCTOK    289    A
COMCTOK  *         THAT MIGHT OCCUR WITH THE HOST.                              COMCTOK    290    A
COMCTOK  *                                                                      COMCTOK    291    A
COMCTOK  *----------------------------------------------------------------------COMCTOK    292    A
COMCTOK  *                                                                      COMCTOK    293    A
COMCTOK  *         TEST - *TEST* MODE.                                          COMCTOK    294    A
COMCTOK  *                                                                      COMCTOK    295    A
COMCTOK  *         THE *COMCTOK* INSTALLER CAN SELECT *COMCTOK* TO BE IN *TEST* COMCTOK    296    A
COMCTOK  *         MODE, WHICH FORCES *COMCTOK* TO HELP US DEBUG OUR INTERFACE  COMCTOK    297    A
COMCTOK  *         TO *COMCTOK* AND ALSO HELP US TO DEBUG OUR TOGEL PROGRAMS.   COMCTOK    298    A
COMCTOK  *                                                                      COMCTOK    299    A
COMCTOK  *         IF THE SYMBOL *TEST* IS SET .NZ., THEN *COMCTOK* WILL FORCE  COMCTOK    300    A
COMCTOK  *         ALL REGISTERS BACK OUT TO *TOKCOM* (WHERE THEY ARE KEPT      COMCTOK    301    A
COMCTOK  *         WHEN NECESSARY) BEFORE EACH AND EVERY TOGEL BINARY INSTRUCTION COMCTOK  302    A
COMCTOK  *         IS EXECUTED/INTERPRETED.  THIS HELPS PROVIDE AN AUDIT TRAIL  COMCTOK    303    A
COMCTOK  *         OF WHAT IS GOING ON.                                         COMCTOK    304    A
COMCTOK  *                                                                      COMCTOK    305    A
COMCTOK  *         IF THE SYMBOL *TEST* IS SET .NZ., THEN THE INSTALLER ALSO    COMCTOK    306    A
COMCTOK  *         OPTIONALLY HAS THE ABILITY TO INTERFACE *COMCTOK* WITH       COMCTOK    307    A
COMCTOK  *         *COMCIDP* (INTERACTIVE DEBUG PACKAGE) VIA *COMDTOK* (IDP     COMCTOK    308    A
COMCTOK  *         UTILITY FOR *COMCTOK*).                                      COMCTOK    309    A
COMCTOK  *                                                                      COMCTOK    310    A
COMCTOK  *         *COMDTOK* CONTAINS SOME USEFUL DEBUGGING TOOLS THAT INTERFACE COMCTOK    311    A
COMCTOK  *         DIRECTLY (AND EASILY) WITH IDP.  *COMDTOK* PROVIDES THE      COMCTOK    312    A
COMCTOK  *         FOLLOWING COMMANDS, AVAILABLE THRU IDP --                    COMCTOK    313    A
COMCTOK  *                                                                      COMCTOK    314    A
COMCTOK  *                BRTOK XXX    BREAKPOINT WHEN THE TOGEL INSTRUCTION   COMCTOK    315    A
COMCTOK  *                             *XXX* IS NEXT TO BE EXECUTED.  IF *XXX*  COMCTOK    316    A
COMCTOK  *                             IS OMITTED, *COMDTOK* WILL BREAKPOINT    COMCTOK    317    A
COMCTOK  *                             ON ANY AND ALL TOGEL INSTRUCTIONS.       COMCTOK    318    A
COMCTOK  *                                                                      COMCTOK    319    A
COMCTOK  *                TOGEL FWA,LWA,LEN  DEASSEMBLES TOGEL BINARY           COMCTOK    320    A
COMCTOK  *                             INSTRUCTIONS BACK INTO A READABLE FORM.  COMCTOK    321    A
COMCTOK  *                                                                      COMCTOK    322    A
COMCTOK  *                TOKEN FWA,LWA,LEN  DISPLAYS THE TOKEN BUFFER.         COMCTOK    323    A
COMCTOK  *                                                                      COMCTOK    324    A
COMCTOK  *                UBTOK        UNBREAK TOGEL BREAKPOINT SET VIA *BRTOK*. COMCTOK    325    A
COMCTOK  *                                                                      COMCTOK    326    A
COMCTOK  ***********************************************************************COMCTOK    327    A
COMCTOK  *                                                                      COMCTOK    328    A
COMCTOK  *         *TOKCOM*                                                     COMCTOK    329    A
COMCTOK  *                                                                      COMCTOK    330    A
COMCTOK  *         *TOKCOM* IS THE USER/TOK COMMUNICATIONS AREA THAT IS USED    COMCTOK    331    A
COMCTOK  *         THROUGHOUT TOKEN GENERATION TO GET AND SAVE CRITICAL         COMCTOK    332    A
COMCTOK  *         ENTOKENING INFORMATION/REGISTERS.                            COMCTOK    333    A
COMCTOK  *                                                                      COMCTOK    334    A
COMCTOK  *         LOCATIONS IN *TOKCOM* ARE DEFINED VIA ORDINALS THAT TAKE     COMCTOK    335    A
COMCTOK  *         THE FORM --                                                  COMCTOK    336    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCTOK

```
COMCTOK  *                                                          COMCTOK    337    A
COMCTOK  *                TC.XXX                                    COMCTOK    338    A
COMCTOK  *                                                          COMCTOK    339    A
COMCTOK  *         WHERE *TC.* IS A COMMON PREFIX, AND *XXX* IS A UNIQUE ORDINAL  COMCTOK    340    A
COMCTOK  *         NAME.  THESE SYMBOLS ARE DEFINED IN *COMSTOK* (COMMON TOKEN    COMCTOK    341    A
COMCTOK  *         GENERATOR INTERFACE TEXT), AND IT IS **STRONGLY** ADVISED      COMCTOK    342    A
COMCTOK  *         THAT THE USER/INSTALLER MAKE HER *TOKCOM* REFERENCES VIA THESE COMCTOK    343    A
COMCTOK  *         SYMBOLS.  THIS WILL PREVENT ANY STUPID BUGS ASSOCIATED WITH    COMCTOK    344    A
COMCTOK  *         *TOKCOM* CHANGES OR ADDITIONS FROM BURNING YOU.  ALSO, THE     COMCTOK    345    A
COMCTOK  *         LENGTH OF *TOKCOM* (IN WORDS) IS DEFINED VIA THE SYMBOL        COMCTOK    346    A
COMCTOK  *         *L.TCOM* (ALSO DEFINED IN *COMSTOK*).  *TC.XXX* SYMBOLS ARE    COMCTOK    347    A
COMCTOK  *         DESCRIBED BELOW.                                               COMCTOK    348    A
COMCTOK  *                                                          COMCTOK    349    A
COMCTOK  *         *COMCTOK* EXPECTS THAT THE CALLER SET UP *TOKCOM* IN ONE       COMCTOK    350    A
COMCTOK  *         OF TWO WAYS: THE 1ST IS AN *EASY* FORM; AND THE 2ND, A MORE    COMCTOK    351    A
COMCTOK  *         DIFFICULT (ALBEIT, MORE FLEXIBLE) FORM.  THE ORIGINAL VERSION  COMCTOK    352    A
COMCTOK  *         OF *COMCTOK* DID NOT HAVE THE *EASIER* FORM: IT AROSE BECAUSE  COMCTOK    353    A
COMCTOK  *         THE *HARDER* FORM TURNED OUT TO BE TOO HARD, AND OVERLY        COMCTOK    354    A
COMCTOK  *         FLEXIBLE FOR MOST *COMCTOK* APPLICATIONS.                      COMCTOK    355    A
COMCTOK  *                                                          COMCTOK    356    A
COMCTOK  *         IN THE FOLLOWING DISCUSSION, USE OF SYMBOLS OF THE FORM        COMCTOK    357    A
COMCTOK  *         *TC.XXX* IMPLIES THE CONTENTS OF A CELL THAT IS AT THE          COMCTOK    358    A
COMCTOK  *         LOCATION --                                                    COMCTOK    359    A
COMCTOK  *                                                          COMCTOK    360    A
COMCTOK  *                FWA+TC.XXX                                COMCTOK    361    A
COMCTOK  *                                                          COMCTOK    362    A
COMCTOK  *         WHERE *FWA* IS THE FWA OF THE CALLER'S *TOKCOM*.               COMCTOK    363    A
COMCTOK  *                                                          COMCTOK    364    A
COMCTOK  *                                                          COMCTOK    365    A
COMCTOK  *         THE *EASY* FORM --                                             COMCTOK    366    A
COMCTOK  *         ---------------                                                COMCTOK    367    A
COMCTOK  *                                                          COMCTOK    368    A
COMCTOK  *           TC.FWA  = ADDR OF THE WORD CONTAINING THE 1ST CHARACTER TO   COMCTOK    369    A
COMCTOK  *                     ENTOKEN (SOURCE IS IN PACKED, 10 CHARS PER WORD,   COMCTOK    370    A
COMCTOK  *                     FORMAT).                                           COMCTOK    371    A
COMCTOK  *                                                          COMCTOK    372    A
COMCTOK  *                     (TC.FWA) IS NOT ALTERED BY *COMCTOK*.              COMCTOK    373    A
COMCTOK  *                                                          COMCTOK    374    A
COMCTOK  *           TC.FCP  = FIRST CHARACTER POSITION WITHIN (TC.FWA) TO BEGIN  COMCTOK    375    A
COMCTOK  *                     ENTOKENING.  CHARACTERS ARE NUMBERED FROM LEFT-TO- COMCTOK    376    A
COMCTOK  *                     RIGHT, 1 TO 10, WITHIN A WORD AS FOLLOWS --        COMCTOK    377    A
COMCTOK  *                                                          COMCTOK    378    A
COMCTOK  *                         +--+--+--+--+--+--+--+--+--+--+  COMCTOK    379    A
COMCTOK  *                         I 1I 2I 3I 4I 5I 6I 7I 8I 9I10I  COMCTOK    380    A
COMCTOK  *                         +--+--+--+--+--+--+--+--+--+--+  COMCTOK    381    A
COMCTOK  *                                                          COMCTOK    382    A
COMCTOK  *                     FOR EXAMPLE, IN 10HABCDEFGHIJ, *A* IS IN CHARACTER COMCTOK    383    A
COMCTOK  *                     POSITION 1, *B* IS IN CHARACTER POSITION 2, ETC.   COMCTOK    384    A
COMCTOK  *                                                          COMCTOK    385    A
COMCTOK  *                     (TC.FCP) IS NOT ALTERED BY *COMCTOK*.              COMCTOK    386    A
COMCTOK  *                                                          COMCTOK    387    A
COMCTOK  *           TC.LWA  = ADDR OF THE WORD CONTAINING THE LAST CHARACTER TO COMCTOK    388    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCTOK

```
COMCTOK  *                    ENTOKEN.                                          COMCTOK    389    A
COMCTOK  *                                                                      COMCTOK    390    A
COMCTOK  *                    (TC.LWA) IS NOT ALTERED BY *COMCTOK*.             COMCTOK    391    A
COMCTOK  *                                                                      COMCTOK    392    A
COMCTOK  *          TC.LCP  = CHARACTER POSITION WITHIN (TC.LWA) OF LAST CHAR   COMCTOK    393    A
COMCTOK  *                    TO ENTOKEN.  CHARACTERS ARE NUMBERED AS FOR       COMCTOK    394    A
COMCTOK  *                    *TC.FCP* ABOVE.                                   COMCTOK    395    A
COMCTOK  *                                                                      COMCTOK    396    A
COMCTOK  *                    (TC.LCP) IS NOT ALTERED BY *COMCTOK*.             COMCTOK    397    A
COMCTOK  *                                                                      COMCTOK    398    A
COMCTOK  *          TC.P    = ADDR OF NEXT *TOGEL* INSTRUCTION TO EXECUTE/      COMCTOK    399    A
COMCTOK  *                    INTERPRET.                                        COMCTOK    400    A
COMCTOK  *                                                                      COMCTOK    401    A
COMCTOK  *          TC.TBA  = ADDR TO STORE NEXT TOKEN THAT IS GENERATED.       COMCTOK    402    A
COMCTOK  *                                                                      COMCTOK    403    A
COMCTOK  *                    NOTE THAT CALLER **MUST** HAVE ALLOCATED ENOUGH   COMCTOK    404    A
COMCTOK  *                    SPACE TO ACCOMODATE THE **MAXIMUM** NR OF TOKENS  COMCTOK    405    A
COMCTOK  *                    THAT *COMCTOK* CAN GENERATE.  THIS IS TO SAY THAT COMCTOK    406    A
COMCTOK  *                    *TOK* DOES NOT CHECK THE *TOKEN BUFFER* FOR ANY   COMCTOK    407    A
COMCTOK  *                    SORT OF OVERFLOW CONDITION...HE JUST KEEPS        COMCTOK    408    A
COMCTOK  *                    GENERATING TOKENS FOR THE LENGTH SPECIFIED        COMCTOK    409    A
COMCTOK  *                    BY THE CALLER VIA *TC.LWA/TC.LCP* OR *TC.SOL*.    COMCTOK    410    A
COMCTOK  *                                                                      COMCTOK    411    A
COMCTOK  *                    UPDATED UPON HITTING END-OF-LINE (IN *TEST*       COMCTOK    412    A
COMCTOK  *                    MODE, UPDATED BEFORE EVERY *TOGEL* INSTRUCTION).  COMCTOK    413    A
COMCTOK  *                                                                      COMCTOK    414    A
COMCTOK  *          TC.RSF  = .NZ. IF THIS IS A *COMCTOK* RESTART, ELSE .ZR.    COMCTOK    415    A
COMCTOK  *                                                                      COMCTOK    416    A
COMCTOK  *                    THIS OPTION IS FOR CALLERS WHO HAVE TO DEAL WITH  COMCTOK    417    A
COMCTOK  *                    CONTINUATION LINES, AND WOULD LIKE A WAY TO       COMCTOK    418    A
COMCTOK  *                    RESTART *COMCTOK* WHERE IT LEFT OFF ON THE        COMCTOK    419    A
COMCTOK  *                    PRECEDING LINE.                                   COMCTOK    420    A
COMCTOK  *                                                                      COMCTOK    421    A
COMCTOK  *                    (TC.RSF) IS NOT ALTERED BY *COMCTOK*.             COMCTOK    422    A
COMCTOK  *                                                                      COMCTOK    423    A
COMCTOK  *            TC.REST = ADDR OF A USER PROCESSOR FOR *RESTART*.  CAN BE COMCTOK    424    A
COMCTOK  *                    SET VIA *ON RESTART* TOGEL MACRO.                 COMCTOK    425    A
COMCTOK  *                                                                      COMCTOK    426    A
COMCTOK  *                                                                      COMCTOK    427    A
COMCTOK  *         THE *HARD* FORM --                                           COMCTOK    428    A
COMCTOK  *         ---------------                                              COMCTOK    429    A
COMCTOK  *                                                                      COMCTOK    430    A
COMCTOK  *            TC.FWA  = .ZR. TO INDICATE *HARD* FORM.                   COMCTOK    431    A
COMCTOK  *                                                                      COMCTOK    432    A
COMCTOK  *            TC.FCP  = NOT USED.                                       COMCTOK    433    A
COMCTOK  *                                                                      COMCTOK    434    A
COMCTOK  *            TC.LWA  = NOT USED.                                       COMCTOK    435    A
COMCTOK  *                                                                      COMCTOK    436    A
COMCTOK  *            TC.LCP  = NOT USED.                                       COMCTOK    437    A
COMCTOK  *                                                                      COMCTOK    438    A
COMCTOK  *            TC.P    = SAME AS FOR *EASY* FORM.                        COMCTOK    439    A
COMCTOK  *                                                                      COMCTOK    440    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCTOK

```
COMCTOK  *          TC.SOA  = ADDR OF CURRENT SOURCE INPUT (10 CHARS PER WORD)
COMCTOK  *                    WORD.  FOR A 1ST ENTRY, THIS IS THE SAME AS FOR
COMCTOK  *                    *TC.FWA*.
COMCTOK  *
COMCTOK  *          TC.SOX  = CURRENT SOURCE INPUT WORD, WITH NEXT CHARACTER TO
COMCTOK  *                    EXTRACT IN LEFT-MOST *CHAR* BITS (WHERE *CHAR* IS
COMCTOK  *                    THE BIT WIDTH OF A SINGLE CHARACTER, I.E. 6 FOR
COMCTOK  *                    DISPLAY CODE).  THAT IS, FIRST CHAR TO EXTRACT IS
COMCTOK  *                    IN BITS 59 THRU 54 (FOR DPC).
COMCTOK  *
COMCTOK  *                    NOTE THAT *TC.SOA* AND *TC.SOX* NEED NOT *MATCH*
COMCTOK  *                    AS MIGHT BE THE CASE IF SOME USER PRE-TOK
COMCTOK  *                    PROCESSOR HAD, FOR EXAMPLE, STRIPPED OFF A LINE
COMCTOK  *                    LABEL/NUMBER.
COMCTOK  *
COMCTOK  *          TC.SOL  = TOTAL NUMBER OF **BITS** IN THIS SOURCE LINE TO
COMCTOK  *                    ENTOKEN.
COMCTOK  *
COMCTOK  *          TC.SOB  = NR OF BITS **REMAINING** IN (TC.SOX).  I.E. FOR
COMCTOK  *                    A FULL WORD, (TC.SOB) = 60D.
COMCTOK  *
COMCTOK  *          TC.SOC  = CURRENT SOURCE CHARACTER, ELSE .MI. IF NEED TO
COMCTOK  *                    EXTRACT A CHARACTER FROM (TC.SOX) FIRST.
COMCTOK  *
COMCTOK  *          TC.TBA  = SAME AS FOR *EASY* FORM.
COMCTOK  *
COMCTOK  *          TC.TBB  = NR OF BITS REMAINING IN TOKEN PACKING REGISTER.
COMCTOK  *
COMCTOK  *
COMCTOK  *          IF YOU DECIDE THAT YOU WANT OR NEED TO USE THE *HARD* FORM, I
COMCTOK  *          SUGGEST A READING OF *COMCBUB* (BURST/BUILD CHARACTERS WITH
COMCTOK  *          BLANK SQUEEZE) AND *COMCBUN* (BURST/BUILD CHARACTERS WITH NO
COMCTOK  *          BLANK SQUEEZE).  MOST OF THESE *TC.* CELLS ARE REALLY JUST
COMCTOK  *          MATERIALIZATIONS OF *BUB/BUN* REGISTERS...
COMCTOK  *
COMCTOK  ***********************************************************************
COMCTOK   TOFU      SPACE  4,10
COMCTOK  **         TOFU - *TOK* FUNCTIONAL UNIT.
COMCTOK  *
COMCTOK  *
COMCTOK  *          THIS MACRO DEFINES THE ENTRY POINT TO A *TOK* FUNCTIONAL
COMCTOK  *          UNIT (TOFU).  EACH *TOFU* WITHIN *TOK* EXECUTES/INTERPRETS
COMCTOK  *          A SINGLE BINARY *TOGEL* INSTRUCTION.
COMCTOK  *
COMCTOK  *          NOTE - SEE FTN 5 IMS/DECK *LEX* FOR MORE INFORMATION.
COMCTOK  *
COMCTOK  * NAM      TOFU   RAD
COMCTOK  *
COMCTOK  *          ENTRY  NAM = NAME OF THIS *TOFU*
COMCTOK  *                 RAD = RESTART ADDR.  ADDR TO TRANSFER CONTROL
COMCTOK  *                       OF CPU TO WHEN RESTARTING TOKEN GENERATOR.
COMCTOK  *
```

```
          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCTOK

```
COMCTOK  *         EXIT   NONE                                                  COMCTOK    493    A
COMCTOK  *                                                                      COMCTOK    494    A
COMCTOK  *         USES   NONE                                                  COMCTOK    495    A
COMCTOK  *                                                                      COMCTOK    496    A
COMCTOK  *         CALLS  NONE                                                  COMCTOK    497    A
COMCTOK                                                                         COMCTOK    498    A
COMCTOK                                                                         COMCTOK    499    A
COMCTOK            PURGMAC TOFU                                                 COMCTOK    500    A
COMCTOK                                                                         COMCTOK    501    A
COMCTOK            MACRO  TOFU,NAM,RAD                                          COMCTOK    502    A
COMCTOK   +        EQ     RAD TOK=MN                                            COMCTOK    503    A
COMCTOK   NAM      BSS    0                                                     COMCTOK    504    A
COMCTOK  *                                                                      COMCTOK    505    A
COMCTOK            IF     -DEF,QUAL$,3                                          COMCTOK    506    A
COMCTOK            QUAL   *                                                     COMCTOK    507    A
COMCTOK   NAM      BSS    0                                                     COMCTOK    508    A
COMCTOK            QUAL   COMCTOK                                               COMCTOK    509    A
COMCTOK   TOFU     ENDM                                                         COMCTOK    510    A
COMCTOK   FW=TCOM  SPACE  4,10                                                  COMCTOK    511    A
COMCTOK  **        FW=TCOM - FWA OF *TOKCOM*.                                   COMCTOK    512    A
COMCTOK  *                                                                      COMCTOK    513    A
COMCTOK  *                                                                      COMCTOK    514    A
COMCTOK  *         *FW=TCOM* CONTAINS THE FWA OF THE CURRENT *USER/TOK*         COMCTOK    515    A
COMCTOK  *         COMMUNICATIONS AREA, *TOKCOM*.                               COMCTOK    516    A
COMCTOK                                                                         COMCTOK    517    A
COMCTOK                                                                         COMCTOK    518    A
COMCTOK   FW=TCOM  BSSZ   1                                                     COMCTOK    519    A
COMCTOK   TF=      SPACE  4,10                                                  COMCTOK    520    A
COMCTOK  **        TF=XXX - *TOK* MASTER CONTROL FLAGS.                         COMCTOK    521    A
COMCTOK  *                                                                      COMCTOK    522    A
COMCTOK  *                                                                      COMCTOK    523    A
COMCTOK  *         SYMBOLS OF THE FORM *TF=XXX*, WHERE *TF=* IS A COMMON PREFIX COMCTOK    524    A
COMCTOK  *         AND *XXX* IS A FLAG NAME, CONTAIN MISCELLANEOUS CONTROL FLAG COMCTOK    525    A
COMCTOK  *         INFORMATION THAT IS USED DURING TOKEN GENERATION.            COMCTOK    526    A
COMCTOK  *                                                                      COMCTOK    527    A
COMCTOK  *         FLAGS, *XXX*, ARE --                                         COMCTOK    528    A
COMCTOK  *                                                                      COMCTOK    529    A
COMCTOK  *         SQZ  = .NZ. IF LAST CHARACTER ACCESS WAS PERFORMED WITH      COMCTOK    530    A
COMCTOK  *                   BLANK SQUEEZE (*COMCBUB* USED), ELSE               COMCTOK    531    A
COMCTOK  *              = .ZR. IF LAST ACCESS WAS NO BLANK SQUEEZE (COMCBUN).   COMCTOK    532    A
COMCTOK  *                                                                      COMCTOK    533    A
COMCTOK  *                USED SO THAT *TOK* CAN KNOW WHETHER TO ADVANCE        COMCTOK    534    A
COMCTOK  *                THE SOURCE LINE CHARACTER POINTER WITH BLANK SQUEEZE  COMCTOK    535    A
COMCTOK  *                OR NOT.                                               COMCTOK    536    A
COMCTOK                                                                         COMCTOK    537    A
COMCTOK                                                                         COMCTOK    538    A
COMCTOK   TF=SQZ   BSSZ   1                                                     COMCTOK    539    A
COMCTOK                                                                         COMCTOK    540    A
COMCTOK            IFNE   TEST,0,1                                              COMCTOK    541    A
COMCTOK   TF=IDP   BSSZ   1                                                     COMCTOK    542    A
COMCTOK   TOKCOM   SPACE  4,10                                                  COMCTOK    543    A
COMCTOK  **        TOKCOM - *TOK* COMMUNICATIONS AREA.                          COMCTOK    544    A
```

```
         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCTOK

```
COMCTOK  *                                                                  COMCTOK    545    A
COMCTOK  *         *TOKCOM* IS THE *USER/TOK* COMMUNICATIONS AREA.  INFORMATION  COMCTOK    546    A
COMCTOK  *         IS PASSED BACK AND FORTH BETWEEN *TOK* AND ITS CALLER  COMCTOK    547    A
COMCTOK  *         VIA THE LOCATIONS IN *TOKCOM*.                           COMCTOK    548    A
COMCTOK  *                                                                  COMCTOK    549    A
COMCTOK  *         SEE *ENTRY/EXIT* CONDITIONS IN *COMCTOK* PREAMBLE.       COMCTOK    550    A
COMCTOK  *                                                                  COMCTOK    551    A
COMCTOK  *         NOTE --                                                  COMCTOK    552    A
COMCTOK  *                                                                  COMCTOK    553    A
COMCTOK  *         CRITICAL ENTOKENING REGISTERS ARE SAVED HERE WHEN NECESSARY.  COMCTOK    554    A
COMCTOK  *         ONE CAN NOT, HOWEVER, EXPECT SIGHT-UNSEEN THAT THESE SAVED  COMCTOK    555    A
COMCTOK  *         REGISTERS ARE ALWAYS UP-TO-DATE.  *TOK* ONLY USES THIS   COMCTOK    556    A
COMCTOK  *         SAVED AREA WHEN ABSOLUTELY NECESSARY SO THAT ENTOKENING  COMCTOK    557    A
COMCTOK  *         CAN PROCEED AS RAPIDLY AS POSSIBLE.                      COMCTOK    558    A
COMCTOK   TOK      SPACE  4,10                                              COMCTOK    559    A
COMCTOK  **        TOK - TOKEN GENERATOR.                                   COMCTOK    560    A
COMCTOK  *                                                                  COMCTOK    561    A
COMCTOK  *                                                                  COMCTOK    562    A
COMCTOK  *         THIS IS THE TOKEN GENERATOR.  IT CONSISTS OF A MAIN DRIVER,  COMCTOK    563    A
COMCTOK  *         *TOK=MN*, SURROUNDED BY *TOK* FUNCTIONAL UNITS, *TOFU*S.  COMCTOK    564    A
COMCTOK  *                                                                  COMCTOK    565    A
COMCTOK  *         *TOK=MN* EXTRACTS THE NEXT *TOGEL* BINARY INSTRUCTION TO  COMCTOK    566    A
COMCTOK  *         EXECUTE/INTERPRET FROM THE CURRENT *TOM* AND FARMS       COMCTOK    567    A
COMCTOK  *         IT OUT TO THE APPROPRIATE *TOFU*, WHERE THIS *TOFU*      COMCTOK    568    A
COMCTOK  *         WILL DO ITS BIT TOWARD ENTOKENING THE INPUT SOURCE LINE.  COMCTOK    569    A
COMCTOK  *                                                                  COMCTOK    570    A
COMCTOK  *         THIS PROCESS REPEATS ITSELF UNTIL A *TOFU* SENSES AN     COMCTOK    571    A
COMCTOK  *         END-OF-LINE (EOL) CONDITION ON THE INPUT SOURCE LINE.    COMCTOK    572    A
COMCTOK  *         WHEN THIS OCCURS, THE EOL SENSING *TOFU* TRANSFERS       COMCTOK    573    A
COMCTOK  *         CONTROL TO *TOK=EOL* WHERE SOME FINAL PROCESSING OCCURS  COMCTOK    574    A
COMCTOK  *         BEFORE *TOK* EXITS BACK TO ITS CALLER.                   COMCTOK    575    A
COMCTOK  *                                                                  COMCTOK    576    A
COMCTOK  *                                                                  COMCTOK    577    A
COMCTOK  *         ENTRY  (B2) = FWA OF *TOKCOM*, THE *USER/TOK* COMMUNICATIONS  COMCTOK    578    A
COMCTOK  *                       AREA.                                      COMCTOK    579    A
COMCTOK  *                                                                  COMCTOK    580    A
COMCTOK  *                                                                  COMCTOK    581    A
COMCTOK  *         EXIT   NONE                                              COMCTOK    582    A
COMCTOK  *                                                                  COMCTOK    583    A
COMCTOK  *         USES   ALL         (BELIEVE IT)                          COMCTOK    584    A
COMCTOK  *                                                                  COMCTOK    585    A
COMCTOK  *         CALLS  BUB,BUN,CCD,RER,SER,TID(IN *TEST* MODE)           COMCTOK    586    A
COMCTOK                                                                     COMCTOK    587    A
COMCTOK                                                                     COMCTOK    588    A
COMCTOK   TOK      SUBR               ** ENTRY/EXIT **                      COMCTOK    589    A
COMCTOK                                                                     COMCTOK    590    A
COMCTOK  *         INITIALIZATION.                                          COMCTOK    591    A
COMCTOK                                                                     COMCTOK    592    A
COMCTOK            IF     -DEF,B1=1,1                                       COMCTOK    593    A
COMCTOK            SB1    1                                                 COMCTOK    594    A
COMCTOK                                                                     COMCTOK    595    A
COMCTOK            SX6    B2+                                               COMCTOK    596    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCTOK

```
COMCTOK            SA6    FW=TCOM     SAVE (B2) = FWA OF *TOKCOM*                    COMCTOK    597    A
COMCTOK                                                                              COMCTOK    598    A
COMCTOK            SA1    B2+TC.FWA                                                  COMCTOK    599    A
COMCTOK            ZR     X1,TOK2     IF CALLER HAS SET UP *TOKCOM* HIMSELF          COMCTOK    600    A
COMCTOK                                                                              COMCTOK    601    A
COMCTOK  *         HERE IF CALLER IS ASKING US TO SET UP SOURCE INPUT POINTERS.     COMCTOK    602    A
COMCTOK  *                                                                           COMCTOK    603    A
COMCTOK  *         THIS MEANS THAT THE CALLER IS USING THE *FWA/FCP* AND            COMCTOK    604    A
COMCTOK  *         *LWA/LCP* CALLING SEQUENCE.  WE NEED TO SET UP --                COMCTOK    605    A
COMCTOK  *                                                                           COMCTOK    606    A
COMCTOK  *           1. *S0A* = SOURCE ADDR, I.E. NEXT ADDR TO GET A SOURCE WORD.   COMCTOK    607    A
COMCTOK  *                                                                           COMCTOK    608    A
COMCTOK  *           2. *SOX* = CURRENT SOURCE WORD, I.E. THE CONTENTS OF THE       COMCTOK    609    A
COMCTOK  *                      LOCATION POINTED TO BY *SOA* SHIFTED                 COMCTOK    610    A
COMCTOK  *                      APPROPRIATELY SO THAT 1ST CHAR TO EXTRACT IS IN      COMCTOK    611    A
COMCTOK  *                      LEFT-MOST CHAR POSITION (BITS 59-54).                COMCTOK    612    A
COMCTOK  *                                                                           COMCTOK    613    A
COMCTOK  *           3. *SOB* = SOURCE BIT, I.E. BIT POSITION WITHIN *SOX* OF       COMCTOK    614    A
COMCTOK  *                      NEXT CHAR TO EXTRACT.                                COMCTOK    615    A
COMCTOK  *                                                                           COMCTOK    616    A
COMCTOK  *           4. *SOL* = NR OF BITS IN SOURCE LINE TO ENTOKEN.               COMCTOK    617    A
COMCTOK  *                                                                           COMCTOK    618    A
COMCTOK  *           5. *SOC* = CURRENT SOURCE CHAR FROM (SOX), I.E. SET            COMCTOK    619    A
COMCTOK  *                      TO -1 TO INDICATE *NO AVAILABLE CHAR*.               COMCTOK    620    A
COMCTOK                                                                              COMCTOK    621    A
COMCTOK            =A2    A1+TC.FCP-TC.FWA  (X2) = 1ST CHAR POSITION                 COMCTOK    622    A
COMCTOK            SB3    X2                                                         COMCTOK    623    A
COMCTOK            SB4    60D/CHAR    (B4) = NR OF CHARS IN A WORD                   COMCTOK    624    A
COMCTOK                                                                              COMCTOK    625    A
COMCTOK            LE     B3,B0,"BLOWUP"    IF BAD CHAR POSITION                     COMCTOK    626    A
COMCTOK            GT     B3,B4,"BLOWUP"    IF BAD CHAR POSITION                     COMCTOK    627    A
COMCTOK                                                                              COMCTOK    628    A
COMCTOK            SX3    CHAR        (X3) = BIT WIDTH OF 1 CHARACTER                COMCTOK    629    A
COMCTOK            SX4    60D+CHAR                                                   COMCTOK    630    A
COMCTOK            IX6    X3*X2       (X6) = 6 * NR OF CHARS (FOR DPC)               COMCTOK    631    A
COMCTOK            IX2    X4-X6       (X2) = 66 - (6 * NR OF CHARS), FOR DPC         COMCTOK    632    A
COMCTOK            BX6    X1          (X6) = *SOA*                                   COMCTOK    633    A
COMCTOK            LX7    X2          (X7) = *SOB*                                   COMCTOK    634    A
COMCTOK            SA6    B2+TC.SOA                                                  COMCTOK    635    A
COMCTOK            SA7    B2+TC.SOB                                                  COMCTOK    636    A
COMCTOK                                                                              COMCTOK    637    A
COMCTOK            SX6    60D                                                        COMCTOK    638    A
COMCTOK            SA5    X1          (X5) = 1ST WORD TO ENTOKEN                     COMCTOK    639    A
COMCTOK            IX7    X6-X2                                                      COMCTOK    640    A
COMCTOK            SB5    X7          (B5) = 60D - SOB                               COMCTOK    641    A
COMCTOK            LX6    X5,B5       (X6) = 1ST WORD TO ENTOKEN W/ 1ST CHAR TO      COMCTOK    642    A
COMCTOK  *                                  ENTOKEN IN LEFT-MOST CHAR POSITION       COMCTOK    643    A
COMCTOK            SA6    B2+TC.SOX                                                  COMCTOK    644    A
COMCTOK                                                                              COMCTOK    645    A
COMCTOK  *         COMPUTE *SOL*, AND SET *SOC*.                                     COMCTOK    646    A
COMCTOK                                                                              COMCTOK    647    A
COMCTOK            SA3    B2+TC.LWA         (X3) = LWA                               COMCTOK    648    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCTOK

```
COMCTOK            =A4    A3+TC.LCP-TC.LWA  (X4) = LCP, LAST CHAR POSITION          COMCTOK    649    A
COMCTOK            SB3    X4                                                         COMCTOK    650    A
COMCTOK                                                                             COMCTOK    651    A
COMCTOK            LE     B3,B0,"BLOWUP"    IF BAD CHAR POSITION                     COMCTOK    652    A
COMCTOK            GT     B3,B4,"BLOWUP"    IF BAD CHAR POSITION                     COMCTOK    653    A
COMCTOK                                                                             COMCTOK    654    A
COMCTOK            SB3    B2          SAVE (B2)                                      COMCTOK    655    A
COMCTOK            RJ     CCD         COMPUTE CHARACTER DIFFERENCE                   COMCTOK    656    A
COMCTOK            MI     X7,"BLOWUP" IF LAST CHAR IS TO THE LEFT OF 1ST CHAR        COMCTOK    657    A
COMCTOK            MX6    -1                                                         COMCTOK    658    A
COMCTOK            SB2    B3          RESTORE (B2)                                   COMCTOK    659    A
COMCTOK            SA6    B2+TC.SOC   SET TO *NO AVAILABLE CHAR IN (X4)*             COMCTOK    660    A
COMCTOK            SA7    B2+TC.SOL                                                  COMCTOK    661    A
COMCTOK                                                                             COMCTOK    662    A
COMCTOK  *         RESTORE ENTOKENING REGISTERS.                                    COMCTOK    663    A
COMCTOK                                                                             COMCTOK    664    A
COMCTOK   TOK2     BSS    0                                                          COMCTOK    665    A
COMCTOK            RJ     RER         RESTORE ENTOKENING REGISTERS                   COMCTOK    666    A
COMCTOK            SA2    B2+TC.TBA                                                  COMCTOK    667    A
COMCTOK            SA3    B2+TC.RSF                                                  COMCTOK    668    A
COMCTOK            =X6    "0."BOS                                                    COMCTOK    669    A
COMCTOK            NZ     X3,TOK=RS   IF THIS IS A RESTART                           COMCTOK    670    A
COMCTOK            SA6    X2          INITIALIZE TOKEN BUFFER W/ *BOS* TOKEN         COMCTOK    671    A
COMCTOK   TOK=MN   SPACE  4,10                                                      COMCTOK    672    A
COMCTOK  **        MAIN LOOP NODE.                                                  COMCTOK    673    A
COMCTOK  *                                                                          COMCTOK    674    A
COMCTOK  *         HERE TO EXTRACT NEXT *TOGEL* BINARY INSTRUCTION FROM *TOM*       COMCTOK    675    A
COMCTOK  *         AND TRANSFER CONTROL TO APPROPRIATE *TOFU*.                      COMCTOK    676    A
COMCTOK  *                                                                          COMCTOK    677    A
COMCTOK  *         FIRST, HOWEVER, NEED TO ASSURE THAT (X4) CONTAINS                COMCTOK    678    A
COMCTOK  *         A USEABLE CHARACTER.                                             COMCTOK    679    A
COMCTOK  *                                                                          COMCTOK    680    A
COMCTOK  *         IF IN *TEST* MODE --                                             COMCTOK    681    A
COMCTOK  *                                                                          COMCTOK    682    A
COMCTOK  *           1. SAVE CRITICAL ENTOKENING REGISTERS IN THEIR APPROPRIATE     COMCTOK    683    A
COMCTOK  *              *TOKCOM* LOCATION TO PROVIDE AN AUDIT TRAIL.                COMCTOK    684    A
COMCTOK  *                                                                          COMCTOK    685    A
COMCTOK  *           2. CALL *TID* (TOK INTERACTIVE DEBUG INTERFACE,                COMCTOK    686    A
COMCTOK  *              IN COMDECK *COMDTOK*) IF *TF=IDP* HAS BEEN SET.             COMCTOK    687    A
COMCTOK                                                                             COMCTOK    688    A
COMCTOK                                                                             COMCTOK    689    A
COMCTOK   TOK=MN   BSS    0           ** MAIN LOOP NODE **                          COMCTOK    690    A
COMCTOK            MX0    -CHAR                                                      COMCTOK    691    A
COMCTOK            PL     X4,TOK4     IF (X4) ALREADY CONTAINS A USEABLE CHAR        COMCTOK    692    A
COMCTOK            BX1    X1-X1       (X1) = CHAR SHIFT MASK (NO CHARS IN GROUP)     COMCTOK    693    A
COMCTOK            SA2    TF=SQZ                                                     COMCTOK    694    A
COMCTOK            MX6    0           CLEAR PACKING REGISTER                         COMCTOK    695    A
COMCTOK            NZ     X2,TOK3     IF BU/BU WITH BLANK SQUEEZE                    COMCTOK    696    A
COMCTOK            SB3    BN.TOCP+BN.TOCL-CHAR                                       COMCTOK    697    A
COMCTOK            RJ     =XBUN       BURST/BUILD WITH NO BLANK SQUEEZE              COMCTOK    698    A
COMCTOK            GT     B7,B0,TOK4  IF NO EOL ENCOUNTERED                          COMCTOK    699    A
COMCTOK            EQ     TOK=EOL                                                    COMCTOK    700    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCTOK

```
COMCTOK                                                                          COMCTOK      701     A
COMCTOK   TOK3     SB3    BB.TOCP+BB.TOCL-CHAR                                   COMCTOK      702     A
COMCTOK            RJ     =XBUB       BURST/BUILD WITH BLANK SQUEEZE             COMCTOK      703     A
COMCTOK            GT     B7,B0,TOK4  IF NO EOL ENCOUNTERED                      COMCTOK      704     A
COMCTOK            EQ     TOK=EOL                                                COMCTOK      705     A
COMCTOK                                                                          COMCTOK      706     A
COMCTOK   TOK4     SA2    A0          (X2) = NEXT TOGEL BINARY INST TO XEQ       COMCTOK      707     A
COMCTOK            SA0    A0+B1                                                  COMCTOK      708     A
COMCTOK                                                                          COMCTOK      709     A
COMCTOK   .T       IFNE   TEST,0                                                 COMCTOK      710     A
COMCTOK            RJ     SER         SAVE ENTOKENING REGISTERS                  COMCTOK      711     A
COMCTOK            SA1    TF=IDP                                                 COMCTOK      712     A
COMCTOK            ZR     X1,TOK5     IF NO IDP INTERFACE SELECTED               COMCTOK      713     A
COMCTOK                                                                          COMCTOK      714     A
COMCTOK            SB2    =YTOK=IDP                                              COMCTOK      715     A
COMCTOK            MI     B2,TOK5     IF USER DID NOT PROVIDE *TOK=IDP*          COMCTOK      716     A
COMCTOK            RJ     =YTOK=IDP   *TOK* INTERACTIVE DEBUG INTERFACE          COMCTOK      717     A
COMCTOK                                                                          COMCTOK      718     A
COMCTOK   TOK5     BSS    0                                                      COMCTOK      719     A
COMCTOK   .T       ENDIF                                                         COMCTOK      720     A
COMCTOK                                                                          COMCTOK      721     A
COMCTOK            LX2    0-TG.TOFUP                                             COMCTOK      722     A
COMCTOK            ERRNZ  TG.TOFUL-18                                            COMCTOK      723     A
COMCTOK            SB2    X2          (B2) = TOFU ADDR FOR THIS INSTRUCTION      COMCTOK      724     A
COMCTOK            JP     FW.TOK+B2                                              COMCTOK      725     A
COMCTOK   TOK=RS   SPACE  4,10                                                   COMCTOK      726     A
COMCTOK   **       TOK=RS - *TOK* RESTART NODE.                                  COMCTOK      727     A
COMCTOK   *                                                                      COMCTOK      728     A
COMCTOK   *        HERE IF CALLER SPECIFIED *RESTART*, I.E. (TC=RSF) .NZ.,       COMCTOK      729     A
COMCTOK   *        AS WOULD COMMONLY OCCUR WHEN A HOST WAS USING *TOK* TO        COMCTOK      730     A
COMCTOK   *        ENTOKEN MULTIPLE SOURCE LINES AS A SINGLE SOURCE STMT.        COMCTOK      731     A
COMCTOK   *                                                                      COMCTOK      732     A
COMCTOK   *        IN THIS CASE, TOKEN GENERATION IS INTERRUPTED AT THE          COMCTOK      733     A
COMCTOK   *        END OF EACH SOURCE LINE AND NEEDS TO BE RESUMED/RESTARTED     COMCTOK      734     A
COMCTOK   *        FOR EACH NEW LINE THAT IS FED TO *TOK*.                       COMCTOK      735     A
COMCTOK   *                                                                      COMCTOK      736     A
COMCTOK   *        CONSIDER --                                                   COMCTOK      737     A
COMCTOK   *                                                                      COMCTOK      738     A
COMCTOK   *          COL 1    6                                                  COMCTOK      739     A
COMCTOK   *                    CRAWLED = OUT OF * A                              COMCTOK      740     A
COMCTOK   *                  + WELL                                              COMCTOK      741     A
COMCTOK   *                                                                      COMCTOK      742     A
COMCTOK   *        LINE 1 OF THIS STMT ENDS WITH THE TOKEN -A-.  WHEN LINE 2     COMCTOK      743     A
COMCTOK   *        IS ACQUIRED, *TOK* HAS TO BE RESTARTED SO THAT THE            COMCTOK      744     A
COMCTOK   *        TOKEN -AWELL- CAN BE GENERATED.                               COMCTOK      745     A
COMCTOK                                                                          COMCTOK      746     A
COMCTOK                                                                          COMCTOK      747     A
COMCTOK   TOK=RS   BSS    0                                                      COMCTOK      748     A
COMCTOK                                                                          COMCTOK      749     A
COMCTOK   *        SET UP (A6).  NEXT ADDR TO STORE TOKEN IS AT (A6)+1.          COMCTOK      750     A
COMCTOK                                                                          COMCTOK      751     A
COMCTOK    +       SA1    X2-1                                                   COMCTOK      752     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCTOK

```
COMCTOK            BX6    X1                                                  COMCTOK    753    A
COMCTOK            SA6    A1          RESTORE (A6)                            COMCTOK    754    A
COMCTOK                                                                       COMCTOK    755    A
COMCTOK  *         RESTORE *TOGEL* BINARY INSTRUCTION WE WERE                  COMCTOK    756    A
COMCTOK  *         EXECUTING WHEN INTERRUPTED.                                 COMCTOK    757    A
COMCTOK                                                                       COMCTOK    758    A
COMCTOK            SA1    B2+TC.REST                                          COMCTOK    759    A
COMCTOK            SA2    A0-B1                                               COMCTOK    760    A
COMCTOK            LX2    0-TG.TOFUP                                          COMCTOK    761    A
COMCTOK            MX0    -CHAR                                               COMCTOK    762    A
COMCTOK            SB4    X2-1        (B4) = TOFU RESTART ADDR                COMCTOK    763    A
COMCTOK            ZR     X1,TOK5A    IF NO *ON RESTART* PROCESSING           COMCTOK    764    A
COMCTOK            SB2    FW.TOK+B4   (B2) = ADDR OF *TOK* RESTART NODE       COMCTOK    765    A
COMCTOK            SB4    FW.TOK+X1   (B4) = ADDR OF *RESTART* PROCESSOR      COMCTOK    766    A
COMCTOK                                                                       COMCTOK    767    A
COMCTOK   TOK5A    JP     B4+         BANG...                                 COMCTOK    768    A
COMCTOK   TOK=CAL  SPACE  4,10                                                COMCTOK    769    A
COMCTOK  **        HERE IF *CALT XX*.                                         COMCTOK    770    A
COMCTOK                                                                       COMCTOK    771    A
COMCTOK                                                                       COMCTOK    772    A
COMCTOK   TOK=CAL  TOFU                                                       COMCTOK    773    A
COMCTOK            LX2    0-TG.CADP+TG.TOFUP-0                                COMCTOK    774    A
COMCTOK            ERRNZ  TG.CADL-18                                          COMCTOK    775    A
COMCTOK            SB2    X2          (B2) = *CALT* ADDR                      COMCTOK    776    A
COMCTOK            JP     FW.TOK+B2                                           COMCTOK    777    A
COMCTOK   TOK=COF  SPACE  4,10                                                COMCTOK    778    A
COMCTOK  **        HERE IF *CASEOF (...)*.                                    COMCTOK    779    A
COMCTOK  *                                                                    COMCTOK    780    A
COMCTOK  *         READY TO PROCESS AND ENTOKEN THE CHARACTER IN (X4) WHICH    COMCTOK    781    A
COMCTOK  *         TERMINATED THE LAST BURST/BUILD.                            COMCTOK    782    A
COMCTOK  *                                                                    COMCTOK    783    A
COMCTOK  *         FIRST, THE CHARACTER IS USED AS AN ORDINAL INTO THE CHARACTER COMCTOK  784    A
COMCTOK  *         MAP *TOKEN* TABLE WHICH CONTAINS A TOKEN SKELETON FOR THIS  COMCTOK    785    A
COMCTOK  *         CHARACTER.  IN THE SIMPLEST CASE, THIS TOKEN SKELETON MERELY COMCTOK   786    A
COMCTOK  *         GETS COPIED TO THE TOKEN BUFFER, AND CONTROL PASSES BACK TO COMCTOK    787    A
COMCTOK  *         *TOK=MN*.                                                   COMCTOK    788    A
COMCTOK  *                                                                    COMCTOK    789    A
COMCTOK  *         THIS TOKEN GENERATOR ALSO HAS THE ABILITY TO MAP MULTIPLE   COMCTOK    790    A
COMCTOK  *         TOKENS BACK TO A NEW SINGLE TOKEN.  THIS PROCESS CAN GREATLY COMCTOK   791    A
COMCTOK  *         SIMPLIFY THE PARSING OF MULTIPLE CHARACTER OPERATORS.       COMCTOK    792    A
COMCTOK  *                                                                    COMCTOK    793    A
COMCTOK  *         FOR EXAMPLE, THE FORTRAN EXPONENTIATION OPERATOR ** CAN BE  COMCTOK    794    A
COMCTOK  *         LOOKED AT AS A * TOKEN FOLLOWED BY A * TOKEN.  HOWEVER,     COMCTOK    795    A
COMCTOK  *         THE PARSER CAN BE MORE STRAIGHT-FORWARD IF A ** TOKEN IS    COMCTOK    796    A
COMCTOK  *         GENERATED INSTEAD.                                          COMCTOK    797    A
COMCTOK  *                                                                    COMCTOK    798    A
COMCTOK  *         THE TECHNIQUE USED HERE IS TO LOOK BACK AT THE TOKEN PRECEDING COMCTOK 799    A
COMCTOK  *         THE ONE BEING GENERATED.  FOR THE ** CASE, WHEN A * TOKEN IS COMCTOK   800    A
COMCTOK  *         GENERATED, THE TOKEN PRECEDING THE * IS EXAMINED;  IF FOUND COMCTOK    801    A
COMCTOK  *         TO BE A * TOKEN, A ** TOKEN REPLACES THEM.                  COMCTOK    802    A
COMCTOK  *                                                                    COMCTOK    803    A
COMCTOK  *         THIS ENTIRE PROCESS IS TABLE DRIVEN, WHICH ALLOWS THE       COMCTOK    804    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCTOK

```
COMCTOK  *         *COMCTOK* USER TO DEFINE HER OWN OPERATORS WITHOUT A LOT          COMCTOK    805    A
COMCTOK  *         OF GRIEF  (SEE *COMATOK*).                                        COMCTOK    806    A
COMCTOK  *                                                                           COMCTOK    807    A
COMCTOK  *         THE SYNTAX TABLE IS A BINARY TREE WHERE *NO MATCH* MOVEMENT       COMCTOK    808    A
COMCTOK  *         IS LEFT-TO-RIGHT, AND *MATCH* MOVEMENT IS TOP-TO-BOTTOM.          COMCTOK    809    A
COMCTOK  *                                                                           COMCTOK    810    A
COMCTOK  *         FOR EXAMPLE, ARBITRARILY CONSIDER THE 2 OPERATORS                 COMCTOK    811    A
COMCTOK  *               /*$  AND  #:$                                              COMCTOK    812    A
COMCTOK  *                                                                           COMCTOK    813    A
COMCTOK  *               TOKEN                                                       COMCTOK    814    A
COMCTOK  *                   +---+---+---+                                           COMCTOK    815    A
COMCTOK  *                   + T + 0 + $ +                                           COMCTOK    816    A
COMCTOK  *                   +---+---+---+                                           COMCTOK    817    A
COMCTOK  *                     .                                                     COMCTOK    818    A
COMCTOK  *               TLINK  .   *$                 :$                            COMCTOK    819    A
COMCTOK  *                   +---+---+---+      +---+---+---+                        COMCTOK    820    A
COMCTOK  *                   + T + F + * +  ..> + T + 0 + : +                        COMCTOK    821    A
COMCTOK  *                   +---+---+---+  .   +---+---+---+                        COMCTOK    822    A
COMCTOK  *                     .   .        .     .                                 COMCTOK    823    A
COMCTOK  *                     .   ..........     .                                 COMCTOK    824    A
COMCTOK  *                     .                  .                                 COMCTOK    825    A
COMCTOK  *                     .  /*$             .  #:$                            COMCTOK    826    A
COMCTOK  *                   +---+---+---+      +---+---+---+                        COMCTOK    827    A
COMCTOK  *                   + 0 + 0 + / +      + 0 + 0 + # +                        COMCTOK    828    A
COMCTOK  *                   +---+---+---+      +---+---+---+                        COMCTOK    829    A
COMCTOK  *                                                                           COMCTOK    830    A
COMCTOK  *               WHERE EACH RECTANGLE CONTAINS 3 SQUARES --                  COMCTOK    831    A
COMCTOK  *                                                                           COMCTOK    832    A
COMCTOK  *                 SQUARE 1  CONTAINS THE *MATCH* LINK.  I.E. IF THE         COMCTOK    833    A
COMCTOK  *                           TOKEN WE ARE LOOKING AT IS PRECEDED BY          COMCTOK    834    A
COMCTOK  *                           THE TOKEN IN SQUARE 3, GO TO WHERE SQUARE 1     COMCTOK    835    A
COMCTOK  *                           POINTS.  (0 DENOTES END-OF-CHAIN)               COMCTOK    836    A
COMCTOK  *                 SQUARE 2  CONTAINS THE *NO MATCH* LINK.  I.E. IF THE      COMCTOK    837    A
COMCTOK  *                           TOKEN WE ARE LOOKING AT IS NOT PRECEDED BY      COMCTOK    838    A
COMCTOK  *                           THE TOKEN IN SQUARE 3, GO TO WHERE SQUARE 2     COMCTOK    839    A
COMCTOK  *                           POINTS.                                         COMCTOK    840    A
COMCTOK  *                 SQUARE 3  CONTAINS THE TOKEN VALUE TO TRY AND MATCH.      COMCTOK    841    A
COMCTOK                                                                              COMCTOK    842    A
COMCTOK                                                                              COMCTOK    843    A
COMCTOK   TOK=COF  TOFU   TOK=MN                                                     COMCTOK    844    A
COMCTOK                                                                              COMCTOK    845    A
COMCTOK  *         CHECK TO SEE IF CHARACTER TO GENERATE TOKEN FOR IS                COMCTOK    846    A
COMCTOK  *         IN *CASEOF* RANGE.  IF SO, USE CHAR AS AN ORDINAL INTO            COMCTOK    847    A
COMCTOK  *         THE *TOKEN* TABLE TO PICK UP TOKEN SKELETON FOR THIS CHAR.        COMCTOK    848    A
COMCTOK  *         IF NOT, PICK UP /ELSC/ *TOKEN* ENTRY.                             COMCTOK    849    A
COMCTOK  *                                                                           COMCTOK    850    A
COMCTOK  *         IF THE CHARACTER IS WITHIN *CASEOF* RANGE, THEN CHAR IS           COMCTOK    851    A
COMCTOK  *         CONVERTED TO A *ROTATED CHARACTER SET*, CALLED *RCS*,             COMCTOK    852    A
COMCTOK  *         WHERE THE 1ST CHARACTER IN *RCS* IS THE 1ST CHARACTER             COMCTOK    853    A
COMCTOK  *         IN *CASEOF* RANGE, *COA*, (BINARY VALUE = 00B).  THE              COMCTOK    854    A
COMCTOK  *         FORMULA FOR CONVERTING CHAR TO ENTOKEN, *CTE*, TO *RCS*           COMCTOK    855    A
COMCTOK  *         IS --                                                             COMCTOK    856    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCTOK

```
COMCTOK  *                                                                  COMCTOK   857   A
COMCTOK  *                RCS = CTE+10000B-COA                              COMCTOK   858   A
COMCTOK  *                                                                  COMCTOK   859   A
COMCTOK  *         IF THE HOST CHARACTER SET CONSISTS OF *CHAR* BIT CHARACTERS,  COMCTOK   860   A
COMCTOK  *         THEN THE ROTATED CHARACTER SET VALUE IS THE LOWER *CHAR*  COMCTOK   861   A
COMCTOK  *         BITS OF *RCS* ABOVE.                                     COMCTOK   862   A
COMCTOK  *                                                                  COMCTOK   863   A
COMCTOK  *         FOR EXAMPLE, CONSIDER --                                 COMCTOK   864   A
COMCTOK  *                                                                  COMCTOK   865   A
COMCTOK  *                CASEOF (+..:)                                     COMCTOK   866   A
COMCTOK  *                                                                  COMCTOK   867   A
COMCTOK  *         THE *RCS* VALUES FOR THIS *CASEOF* RANGE (USING DISPLAY  COMCTOK   868   A
COMCTOK  *         CODE, 6 BIT, CHARACTERS) ARE --                          COMCTOK   869   A
COMCTOK  *                                                                  COMCTOK   870   A
COMCTOK  *                +      RCS = 45B+10000B-45B                       COMCTOK   871   A
COMCTOK  *                           = 10000B                               COMCTOK   872   A
COMCTOK  *                                                                  COMCTOK   873   A
COMCTOK  *                       EXTRACT LOWER 6 BITS --  RCS = 00B         COMCTOK   874   A
COMCTOK  *                                                                  COMCTOK   875   A
COMCTOK  *                -      RCS = 46B+10000B-45B                       COMCTOK   876   A
COMCTOK  *                           = 10001B                               COMCTOK   877   A
COMCTOK  *                                                                  COMCTOK   878   A
COMCTOK  *                       EXTRACT LOWER 6 BITS --  RCS = 01B         COMCTOK   879   A
COMCTOK  *                                                                  COMCTOK   880   A
COMCTOK  *                SKIP TO                                           COMCTOK   881   A
COMCTOK  *                                                                  COMCTOK   882   A
COMCTOK  *                ;      RCS = 77B+10000B-45B                       COMCTOK   883   A
COMCTOK  *                           = 10032B                               COMCTOK   884   A
COMCTOK  *                                                                  COMCTOK   885   A
COMCTOK  *                       EXTRACT LOWER 6 BITS --  RCS = 32B         COMCTOK   886   A
COMCTOK  *                                                                  COMCTOK   887   A
COMCTOK  *                :      RCS = 00B+10000B-45B                       COMCTOK   888   A
COMCTOK  *                           = 7733B                                COMCTOK   889   A
COMCTOK  *                                                                  COMCTOK   890   A
COMCTOK  *                       EXTRACT LOWER 6 BITS --  RCS = 33B         COMCTOK   891   A
COMCTOK  *                                                                  COMCTOK   892   A
COMCTOK  *         IT IS EASY TO SEE NOW THAT THE *RCS* VALUES ARE ALSO     COMCTOK   893   A
COMCTOK  *         *TOKEN* ORDINALS.                                        COMCTOK   894   A
COMCTOK                                                                     COMCTOK   895   A
COMCTOK                                                                     COMCTOK   896   A
COMCTOK            LX2    0-TG.COAP+TG.TOFUP-0                              COMCTOK   897   A
COMCTOK            BX6    -X0*X2      EXTRACT *COA*                         COMCTOK   898   A
COMCTOK            LX2    0-TG.COZP+TG.COAP-0                               COMCTOK   899   A
COMCTOK            BX7    -X0*X2      EXTRACT *COZ*                         COMCTOK   900   A
COMCTOK            SX1    X4+10000B                                         COMCTOK   901   A
COMCTOK            LX2    0-TG.TADP+TG.COZP-0                               COMCTOK   902   A
COMCTOK            IX3    X1-X6                                             COMCTOK   903   A
COMCTOK            ERRNZ  TG.TADL-18                                        COMCTOK   904   A
COMCTOK            =B3    FW.TOK+X2   (B3) = FWA OF *TOKEN* CHAR MAP        COMCTOK   905   A
COMCTOK  *                                  FOR THIS *CASEOF*              COMCTOK   906   A
COMCTOK            BX1    -X0*X3      (X1) = *RCS* REPRESENTATION OF CHAR   COMCTOK   907   A
COMCTOK  *                                  TO GENERATE TOKEN FOR          COMCTOK   908   A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCTOK

```
COMCTOK            IX6    X7-X1                                                COMCTOK    909    A
COMCTOK            SB2    X1          (B2) = ORD OF *TOKEN* SKEL FOR THIS CHAR COMCTOK    910    A
COMCTOK            SB5    A6          INITIALIZE (B5) = ADDR OF PRECEDING TOKEN COMCTOK   911    A
COMCTOK            PL     X6,TOK6     IF CHAR IS WITHIN *CASEOF* RANGE         COMCTOK    912    A
COMCTOK            SB2    X7+B1       (B2) = ORD OF /ELSC/ *TOKEN* SKEL        COMCTOK    913    A
COMCTOK                                                                        COMCTOK    914    A
COMCTOK   TOK6     SA1    B3+B2       (X1) = *TOKEN* SKEL FOR THIS CHAR        COMCTOK    915    A
COMCTOK                                                                        COMCTOK    916    A
COMCTOK   *        INITIALIZE FOR *TOKEN/TLINK* SYNTAX CHECKING.               COMCTOK    917    A
COMCTOK                                                                        COMCTOK    918    A
COMCTOK            ERRNZ  TK.TOTP-0                                            COMCTOK    919    A
COMCTOK            ERRNZ  TK.TOTL-18                                           COMCTOK    920    A
COMCTOK            ERRNZ  TK.PADL-18                                           COMCTOK    921    A
COMCTOK            ERRNZ  TK.LNKL-18                                           COMCTOK    922    A
COMCTOK            SB5    A6          INITIALIZE (B5) = ADDR OF PRECEDING TOKEN COMCTOK   923    A
COMCTOK            SX6    X1                                                  COMCTOK    924    A
COMCTOK            LX1    0-TK.PADP                                           COMCTOK    925    A
COMCTOK            SX0    X1          (X0) = ADDR OF USER PROCESSING ADDR FOR COMCTOK    926    A
COMCTOK   *                                THIS TOKEN, ELSE .ZR. IF NONE       COMCTOK    927    A
COMCTOK            LX1    0-TK.LNKP+TK.PADP-0                                 COMCTOK    928    A
COMCTOK            NO                                                          COMCTOK    929    A
COMCTOK            =B4    FW.TOK+X1   (B4) = ADDR OF 1ST *TLINK* ENTRY, ELSE  COMCTOK    930    A
COMCTOK   *                                .ZR. IF NONE                        COMCTOK    931    A
COMCTOK   TOK=COF  SPACE  4,10                                                 COMCTOK    932    A
COMCTOK   **       THIS IS THE MAIN LOOP FOR MOVEMENT THRU *TLINK*, THE        COMCTOK    933    A
COMCTOK   *        MULTIPLE CHARACTER OPERATOR SYNTAX TREE --                  COMCTOK    934    A
COMCTOK   *                                                                    COMCTOK    935    A
COMCTOK   *          1. IF HAVE JUST FOUND A *TRUE* TOKEN PAIR, THEN WE WOULD  COMCTOK    936    A
COMCTOK   *             HAVE MOVED *DOWN* THE TREE  (SEE DIAGRAM).             COMCTOK    937    A
COMCTOK   *                                                                    COMCTOK    938    A
COMCTOK   *          2. IF HAVE JUST ENCOUNTERED A *FALSE* TOKEN PAIR, THEN    COMCTOK    939    A
COMCTOK   *             WE WOULD HAVE MOVED *RIGHT* IN THE TREE.               COMCTOK    940    A
COMCTOK   *                                                                    COMCTOK    941    A
COMCTOK   *        ENTRY  (B4) = ADDR OF NEXT *TLINK* ENTRY  (EITHER *DOWN* OR COMCTOK    942    A
COMCTOK   *                     *RIGHT* OF CURRENT ENTRY), ELSE                COMCTOK    943    A
COMCTOK   *                   = .ZR. IF CURRENT ENTRY WAS END-OF-CHAIN         COMCTOK    944    A
COMCTOK   *                (B5) = ADDR OF NEXT TOKEN TO CHECK                  COMCTOK    945    A
COMCTOK   *                                                                    COMCTOK    946    A
COMCTOK   *        LOCKED X - 0,4,5,6                                          COMCTOK    947    A
COMCTOK   *               A - 0,4,5,6                                          COMCTOK    948    A
COMCTOK   *               B - 6,7                                              COMCTOK    949    A
COMCTOK                                                                        COMCTOK    950    A
COMCTOK            ERRNZ  TL.TRUL-18                                           COMCTOK    951    A
COMCTOK            ERRNZ  TL.FALL-18                                           COMCTOK    952    A
COMCTOK            ERRNZ  TL.PREL-18                                           COMCTOK    953    A
COMCTOK                                                                        COMCTOK    954    A
COMCTOK   TOK7     ZR     B4,TOK8A    IF END-OF-CHAIN ENCOUNTERED              COMCTOK    955    A
COMCTOK            SA1    B4          (A1,X1) = A+C OF 1ST WORD OF TLINK ENTRY COMCTOK    956    A
COMCTOK            SA2    B5          (A2,X2) = A+C OF PRECEDING TOKEN         COMCTOK    957    A
COMCTOK            LX1    0-TL.PREP                                           COMCTOK    958    A
COMCTOK            SB2    X1          (B2) = O.PRE                            COMCTOK    959    A
COMCTOK            SB3    X2          (B3) = O.XXX                            COMCTOK    960    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCTOK

```
COMCTOK            LX1    0-TL.FALP                                          COMCTOK    961    A
COMCTOK            =B4    FW.TOK+X1   (B4) = ADDR OF NEXT TLINK ENTRY IN     COMCTOK    962    A
COMCTOK  *                             *FALSE* CHAIN                        COMCTOK    963    A
COMCTOK            NE     B2,B3,TOK7  IF *O.XXX* NOT PRECEDED BY *O.PRE*     COMCTOK    964    A
COMCTOK            LX1    0-TL.TRUP+TL.FALP-0                                COMCTOK    965    A
COMCTOK            =B3    FW.TOK+X1   (B3) = ADDR OF NEXT TLINK ENTRY IN     COMCTOK    966    A
COMCTOK  *                             *TRUE* CHAIN                         COMCTOK    967    A
COMCTOK                                                                     COMCTOK    968    A
COMCTOK  *         HERE IF <O.PRE><O.XXX> OCCURRED.                         COMCTOK    969    A
COMCTOK                                                                     COMCTOK    970    A
COMCTOK            LX1    59-TL.MAFP+TL.TRUP-0                               COMCTOK    971    A
COMCTOK            PL     X1,TOK8     IF EXACT CHARACTER MATCH NOT NEEDED    COMCTOK    972    A
COMCTOK                                                                     COMCTOK    973    A
COMCTOK  *         HERE IF AN EXACT CHARACTER MATCH REQUIRED.               COMCTOK    974    A
COMCTOK                                                                     COMCTOK    975    A
COMCTOK            SA1    A1+B1       (X1) = CHARS TO CHECK FOR, -L- FMT     COMCTOK    976    A
COMCTOK            MX7    TL.MACL                                            COMCTOK    977    A
COMCTOK            ERRNZ  TL.MACP+TL.MACL-60D                                COMCTOK    978    A
COMCTOK            BX2    X7*X2       (X2) = CHARS TO CHECK, -L- FMT         COMCTOK    979    A
COMCTOK            NO                                                        COMCTOK    980    A
COMCTOK            BX7    X1-X2                                              COMCTOK    981    A
COMCTOK            NZ     X7,TOK7     IF NO MATCH                            COMCTOK    982    A
COMCTOK                                                                     COMCTOK    983    A
COMCTOK   TOK8     SB4    B3                                                 COMCTOK    984    A
COMCTOK            SB5    B5-B1                                              COMCTOK    985    A
COMCTOK            NZ     B3,TOK7     IF NEED TO TRY NEXT *TRUE* TLINK ENTRY COMCTOK    986    A
COMCTOK            SA1    A1+B1       (X1) = REPLACEMENT TOKEN               COMCTOK    987    A
COMCTOK            SX6    X1                                                 COMCTOK    988    A
COMCTOK            LX1    0-TK.PADP                                          COMCTOK    989    A
COMCTOK            SX0    X1                                                 COMCTOK    990    A
COMCTOK            LX1    0-TK.LNKP+TK.PADP-0                                COMCTOK    991    A
COMCTOK            =B4    FW.TOK+X1                                          COMCTOK    992    A
COMCTOK            EQ     TOK9                                               COMCTOK    993    A
COMCTOK                                                                     COMCTOK    994    A
COMCTOK  *         HERE WHEN END-OF-CHAIN ENCOUNTERED -- TIME TO INVOKE     COMCTOK    995    A
COMCTOK  *         ANY SPECIAL PROCESSING THAT MIGHT BE REQUIRED FOR TOKEN  COMCTOK    996    A
COMCTOK  *         THAT IS TO BE GENERATED.                                 COMCTOK    997    A
COMCTOK                                                                     COMCTOK    998    A
COMCTOK   TOK8A    SB5    A6+         RESET (B5) = ADDR WE ORIGINALLY BEGAN AT COMCTOK  999    A
COMCTOK                                                                     COMCTOK   1000    A
COMCTOK   TOK9     SB3    FW.TOK+X0   (B3) = ADDR OF SPECIAL PROCESSOR, ELSE COMCTOK   1001    A
COMCTOK  *                             .ZR. IF NONE                         COMCTOK   1002    A
COMCTOK            ZR     B3,TOK10    IF NO SPECIAL PROCESSOR                COMCTOK   1003    A
COMCTOK            MI     B3,TOK9A    IF MERELY XFERRING WITHIN *TOM*        COMCTOK   1004    A
COMCTOK            JP     B3+                                                COMCTOK   1005    A
COMCTOK                                                                     COMCTOK   1006    A
COMCTOK   TOK9A    SA0    -B3         RESET -P- REGISTER                     COMCTOK   1007    A
COMCTOK                                                                     COMCTOK   1008    A
COMCTOK  *         RETURN FROM SPECIAL PROCESSING -- ENTER NEW TOKEN        COMCTOK   1009    A
COMCTOK  *         INTO TOKEN BUFFER.                                       COMCTOK   1010    A
COMCTOK                                                                     COMCTOK   1011    A
COMCTOK   TOK=COS  BSS    0           ** RETURN WITH TOKEN STORE **        COMCTOK   1012    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCTOK

```
COMCTOK   TOK10    SA6    B5+1                                                  COMCTOK    1013    A
COMCTOK                                                                         COMCTOK    1014    A
COMCTOK   TOK=CON  BSS    0           ** RETURN WITH NO TOKEN STORE **          COMCTOK    1015    A
COMCTOK   TOK11    NZ     B4,TOK7     IF NEED TO CHECK LINK FOR NEW TOKEN       COMCTOK    1016    A
COMCTOK            SX4    -1          SET TO *NO USEABLE CHAR IN (X4)*          COMCTOK    1017    A
COMCTOK            EQ     TOK=MN                                                COMCTOK    1018    A
COMCTOK   TOK=ELS  SPACE  4,10                                                  COMCTOK    1019    A
COMCTOK  **        HERE IF *ELST*.                                              COMCTOK    1020    A
COMCTOK  *                                                                      COMCTOK    1021    A
COMCTOK  *         NOTE: *TEST* MODE ONLY.                                      COMCTOK    1022    A
COMCTOK                                                                         COMCTOK    1023    A
COMCTOK   .T       IFNE   TEST,0                                                COMCTOK    1024    A
COMCTOK   TOK=ELS  TOFU                                                         COMCTOK    1025    A
COMCTOK            EQ     TOK=MN                                                COMCTOK    1026    A
COMCTOK   .T       ENDIF                                                        COMCTOK    1027    A
COMCTOK   TOK=ENT  SPACE  4,10                                                  COMCTOK    1028    A
COMCTOK  **        HERE IF *ENDT*.                                              COMCTOK    1029    A
COMCTOK  *                                                                      COMCTOK    1030    A
COMCTOK  *         NOTE: *TEST* MODE ONLY.                                      COMCTOK    1031    A
COMCTOK                                                                         COMCTOK    1032    A
COMCTOK   .T       IFNE   TEST,0                                                COMCTOK    1033    A
COMCTOK   TOK=ENT  TOFU                                                         COMCTOK    1034    A
COMCTOK            EQ     TOK=MN                                                COMCTOK    1035    A
COMCTOK   .T       ENDIF                                                        COMCTOK    1036    A
COMCTOK   TOK=EOL  SPACE  4,10                                                  COMCTOK    1037    A
COMCTOK  **        HERE WHEN EOL ENCOUNTERED.                                   COMCTOK    1038    A
COMCTOK  *                                                                      COMCTOK    1039    A
COMCTOK  *           1. SAVE CRITICAL ENTOKENING REGISTERS SO THAT WE CAN       COMCTOK    1040    A
COMCTOK  *              RECOVER IF NEXT LINE IS CONTINUATION OF THIS LINE.      COMCTOK    1041    A
COMCTOK  *                                                                      COMCTOK    1042    A
COMCTOK  *           2. IF *ON EOL,XXX* INDICATED THAT CALLER WISHES            COMCTOK    1043    A
COMCTOK  *              TO PERFORM SOME END-OF-LINE PROCESSING, THEN GO DO IT.  COMCTOK    1044    A
COMCTOK                                                                         COMCTOK    1045    A
COMCTOK   TOK=EOL  BSS    0           ** ENTRY **                               COMCTOK    1046    A
COMCTOK            SX6    "O."EOS                                               COMCTOK    1047    A
COMCTOK            MX4    -1          SET TO *NO USEABLE CHAR IN (X4)*          COMCTOK    1048    A
COMCTOK            SA6    A6+B1       MARK EOS                                  COMCTOK    1049    A
COMCTOK            RJ     SER         SAVE ENTOKENING REGISTERS                 COMCTOK    1050    A
COMCTOK            SA1    B2+TC.EOL                                             COMCTOK    1051    A
COMCTOK            ZR     X1,EXIT.    IF NO SPECIAL PROCESSING                  COMCTOK    1052    A
COMCTOK            SB2    X1                                                    COMCTOK    1053    A
COMCTOK            JP     B2          DO WHAT YOU WILL...                       COMCTOK    1054    A
COMCTOK   TOK=GEN  SPACE  4,10                                                  COMCTOK    1055    A
COMCTOK  **        HERE IF *GENTOK TOT*.                                        COMCTOK    1056    A
COMCTOK                                                                         COMCTOK    1057    A
COMCTOK                                                                         COMCTOK    1058    A
COMCTOK   TOK=GEN  TOFU                                                         COMCTOK    1059    A
COMCTOK            LX2    0-TG.GTTP+TG.TOFUP-0                                  COMCTOK    1060    A
COMCTOK            ERRNZ  TG.GTTL-18                                            COMCTOK    1061    A
COMCTOK            SX6    X2          (X6) = TOKEN TYPE TO GENERATE             COMCTOK    1062    A
COMCTOK            MX4    -1          SET TO *NO USEABLE CHAR IN (X4)*          COMCTOK    1063    A
COMCTOK            SA6    A6+B1                                                 COMCTOK    1064    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCTOK

```
COMCTOK            EQ     TOK=MN                                                  COMCTOK    1065    A
COMCTOK   TOK=GN   SPACE  4,10                                                     COMCTOK    1066    A
COMCTOK   **       HERE IF *GROUP,...,NSQZ*.                                       COMCTOK    1067    A
COMCTOK                                                                            COMCTOK    1068    A
COMCTOK   TOK=GN   TOFU   TOK13                                                    COMCTOK    1069    A
COMCTOK            LX2    0-TG.TOTP+TG.TOFUP-0                                    COMCTOK    1070    A
COMCTOK            ERRNZ  TG.TOTL-18                                              COMCTOK    1071    A
COMCTOK            SX6    X2          (X6) = 42/0,18/TOKEN TYPE FOR THIS *GROUP*  COMCTOK    1072    A
COMCTOK            SB3    BN.TOCP+BN.TOCL-CHAR                                    COMCTOK    1073    A
COMCTOK                                                                            COMCTOK    1074    A
COMCTOK   TOK12    LX2    0-TG.MXAP+TG.TOTP-0                                     COMCTOK    1075    A
COMCTOK            SA1    FW.TOK+X2   (X1) = CHAR SHIFT MASK FOR THIS *GROUP*     COMCTOK    1076    A
COMCTOK            BX7    X7-X7                                                   COMCTOK    1077    A
COMCTOK            SA7    TF=SQZ      SET TO *BU/BU WITH NO BLANK SQZ*            COMCTOK    1078    A
COMCTOK            RJ     BUN         BURST/BUILD CHARACTERS WITH NO BLANK SQZ    COMCTOK    1079    A
COMCTOK            GT     B7,B0,TOK=MN IF DID NOT ENCOUNTER END-OF-LINE           COMCTOK    1080    A
COMCTOK            EQ     TOK=EOL                                                 COMCTOK    1081    A
COMCTOK                                                                            COMCTOK    1082    A
COMCTOK   **       HERE FOR *GROUP,...,NSQZ* RESTART.                              COMCTOK    1083    A
COMCTOK                                                                            COMCTOK    1084    A
COMCTOK   TOK13    SB2    BN.TOCP+BN.TOCL-CHAR                                    COMCTOK    1085    A
COMCTOK            EQ     B3,B2,TOK=GN IF LAST TOKEN GENERATED WAS FULL           COMCTOK    1086    A
COMCTOK   +        SA1    A6-B1                                                   COMCTOK    1087    A
COMCTOK            BX7    X6          SAVE (X6)                                   COMCTOK    1088    A
COMCTOK            LX6    X1                                                      COMCTOK    1089    A
COMCTOK            SA6    A1          (A6+1) = ADDR TO (RE)STORE CURRENT          COMCTOK    1090    A
COMCTOK   *                              UNFINISHED TOKEN                          COMCTOK    1091    A
COMCTOK            LX2    0-TG.TOTP+TG.TOFUP-0                                    COMCTOK    1092    A
COMCTOK            BX6    X7          RESTORE (X6) = PARTIALLY FILLED TOKEN       COMCTOK    1093    A
COMCTOK            EQ     TOK12                                                   COMCTOK    1094    A
COMCTOK   TOK=GO   SPACE  4,10                                                     COMCTOK    1095    A
COMCTOK   **       HERE IF *GOTO XX*.                                              COMCTOK    1096    A
COMCTOK                                                                            COMCTOK    1097    A
COMCTOK   TOK=GO   TOFU                                                            COMCTOK    1098    A
COMCTOK            LX2    0-TG.GOTP+TG.TOFUP-0                                    COMCTOK    1099    A
COMCTOK            ERRNZ  TG.GOTL-18                                              COMCTOK    1100    A
COMCTOK            SA0    FW.TOK+X2                                               COMCTOK    1101    A
COMCTOK            EQ     TOK=MN                                                  COMCTOK    1102    A
COMCTOK   TOK=GS   SPACE  4,10                                                     COMCTOK    1103    A
COMCTOK   **       HERE IF *GROUP,...,SQZ*.                                        COMCTOK    1104    A
COMCTOK                                                                            COMCTOK    1105    A
COMCTOK   TOK=GS   TOFU   TOK15                                                    COMCTOK    1106    A
COMCTOK            LX2    0-TG.TOTP+TG.TOFUP-0                                    COMCTOK    1107    A
COMCTOK            ERRNZ  TG.TOTL-18                                              COMCTOK    1108    A
COMCTOK            SX6    X2          (X6) = 42/0,18/TOKEN TYPE FOR THIS *GROUP*  COMCTOK    1109    A
COMCTOK            SB3    BB.TOCP+BB.TOCL-CHAR                                    COMCTOK    1110    A
COMCTOK                                                                            COMCTOK    1111    A
COMCTOK   TOK14    LX2    0-TG.MXAP+TG.TOTP-0                                     COMCTOK    1112    A
COMCTOK            ERRNZ  TG.MXAL-18                                              COMCTOK    1113    A
COMCTOK            SA1    FW.TOK+X2   (X1) = CHAR SHIFT MASK FOR THIS *GROUP*     COMCTOK    1114    A
COMCTOK            SX7    B1                                                      COMCTOK    1115    A
COMCTOK            SA7    TF=SQZ      SET TO *BU/BU WITH BLANK SQZ*               COMCTOK    1116    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCTOK

```
COMCTOK            RJ     BUB         BURST/BUILD WITH BLANK SQUEEZE                    COMCTOK    1117    A
COMCTOK            GT     B7,B0,TOK=MN  IF DID NOT ENCOUNTER END-OF-LINE                COMCTOK    1118    A
COMCTOK            EQ     TOK=EOL                                                       COMCTOK    1119    A
COMCTOK                                                                                 COMCTOK    1120    A
COMCTOK  *         HERE FOR *GROUP,...,SQZ* RESTART.                                    COMCTOK    1121    A
COMCTOK                                                                                 COMCTOK    1122    A
COMCTOK   TOK15    SB2    BB.TOCP+BB.TOCL-CHAR                                          COMCTOK    1123    A
COMCTOK            EQ     B3,B2,TOK=GS IF LAST TOKEN GENERATED WAS FULL                 COMCTOK    1124    A
COMCTOK   +        SA1    A6-B1                                                         COMCTOK    1125    A
COMCTOK            BX7    X6          SAVE (X6)                                         COMCTOK    1126    A
COMCTOK            LX6    X1                                                            COMCTOK    1127    A
COMCTOK            SA6    A1          (A6+1) = ADDR TO (RE)STORE CURRENT                COMCTOK    1128    A
COMCTOK  *                                    UNFINISHED TOKEN                         COMCTOK    1129    A
COMCTOK            LX2    0-TG.TOTP+TG.TOFUP-0                                          COMCTOK    1130    A
COMCTOK            BX6    X7          RESTORE (X6) = PARTIALLY FILLED TOKEN             COMCTOK    1131    A
COMCTOK            EQ     TOK14                                                         COMCTOK    1132    A
COMCTOK   TOK=IFT  SPACE  4,10                                                           COMCTOK    1133    A
COMCTOK  **        HERE IF *IFT (...)*.                                                  COMCTOK    1134    A
COMCTOK                                                                                 COMCTOK    1135    A
COMCTOK   TOK=IFT  TOFU                                                                 COMCTOK    1136    A
COMCTOK            LX2    0-TG.MXAP+TG.TOFUP-0                                          COMCTOK    1137    A
COMCTOK            ERRNZ  TG.MXAL-18                                                    COMCTOK    1138    A
COMCTOK            =A1    FW.TOK+X2   (X1) = CHAR SHIFT MASK FOR THIS *IFT*             COMCTOK    1139    A
COMCTOK            SB2    X4+10000B-BICH                                                COMCTOK    1140    A
COMCTOK            LX2    0-TG.FALP+TG.MXAP-0                                           COMCTOK    1141    A
COMCTOK            LX7    X1,B2                                                         COMCTOK    1142    A
COMCTOK            MI     X7,TOK=MN   IF CHARACTER IS IN THIS GROUP (TRUE)...           COMCTOK    1143    A
COMCTOK            ERRNZ  TG.FALL-18                                                    COMCTOK    1144    A
COMCTOK            SA0    FW.TOK+X2   (A0) = ADDR OF *FALSE* BINARY TOGEL INST          COMCTOK    1145    A
COMCTOK            EQ     TOK=MN                                                        COMCTOK    1146    A
COMCTOK   TOK=ON   SPACE  4,10                                                           COMCTOK    1147    A
COMCTOK  **        HERE IF *ON I,V*.                                                     COMCTOK    1148    A
COMCTOK                                                                                 COMCTOK    1149    A
COMCTOK                                                                                 COMCTOK    1150    A
COMCTOK   TOK=ON   TOFU                                                                 COMCTOK    1151    A
COMCTOK            SA1    FW=TCOM     (X1) = FWA OF *TOKCOM*                            COMCTOK    1152    A
COMCTOK            LX2    0-TG.INVP+TG.TOFUP-0                                          COMCTOK    1153    A
COMCTOK            ERRNZ  TG.INVL-18                                                    COMCTOK    1154    A
COMCTOK            SX7    X2          (X7) = INTERUPT VALUE                             COMCTOK    1155    A
COMCTOK            SB2    X1                                                            COMCTOK    1156    A
COMCTOK            LX2    0-TG.INCP+TG.INVP-0                                           COMCTOK    1157    A
COMCTOK            ERRNZ  TG.INCL-18                                                    COMCTOK    1158    A
COMCTOK            SA7    B2+X2       INTERUPT CELL = INTERUPT VALUE                    COMCTOK    1159    A
COMCTOK            EQ     TOK=MN                                                        COMCTOK    1160    A
COMCTOK   TOK=XEQ  SPACE  4,10                                                           COMCTOK    1161    A
COMCTOK  **        HERE IF *XEQ P*.                                                      COMCTOK    1162    A
COMCTOK                                                                                 COMCTOK    1163    A
COMCTOK   TOK=XEQ  TOFU                                                                 COMCTOK    1164    A
COMCTOK            LX2    0-TG.XADP+TG.TOFUP-0                                          COMCTOK    1165    A
COMCTOK            ERRNZ  TG.XADL-18                                                    COMCTOK    1166    A
COMCTOK            SA1    XEQA        (X1) = *GOTO* SKELETON                            COMCTOK    1167    A
COMCTOK            SX7    A0          (X7) = RETURN ADDR                                COMCTOK    1168    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCTOK

```
COMCTOK            =A0    FW.TOK+X2+1 (A0) = ADDR OF 1ST INST TO XEQ IN PROC          COMCTOK    1169    A
COMCTOK            LX7    TG.GOTP-0                                                   COMCTOK    1170    A
COMCTOK            BX7    X1+X7       (X7) = FABRICATED /GOTO *+1/                    COMCTOK    1171    A
COMCTOK            SA7    A0-B1       PLUG /GOTO *+1/  (JUST LIKE SEYMOUR...)         COMCTOK    1172    A
COMCTOK            EQ     TOK=MN                                                      COMCTOK    1173    A
COMCTOK                                                                               COMCTOK    1174    A
COMCTOK                                                                               COMCTOK    1175    A
COMCTOK   XEQA     GOTO   0           SKELETON FOR /GOTO *+1/ PLUG                    COMCTOK    1176    A
COMCTOK   CCD      SPACE  4,10                                                        COMCTOK    1177    A
COMCTOK  **        CCD - COMPUTE CHARACTER DIFFERENCE.                                COMCTOK    1178    A
COMCTOK  *                                                                            COMCTOK    1179    A
COMCTOK  *                                                                            COMCTOK    1180    A
COMCTOK  *         GIVEN 2 CHARACTER POSITIONS WITHIN A SOURCE LINE, *CCD*            COMCTOK    1181    A
COMCTOK  *         WILL RETURN THE NR OF CHARACTERS THAT LIE BETWEEN THEM.            COMCTOK    1182    A
COMCTOK  *                                                                            COMCTOK    1183    A
COMCTOK  *         EACH CHARACTER POSITION IS REPRESENTED IN *TOK*S WORD+BIT          COMCTOK    1184    A
COMCTOK  *         NOTATION.                                                          COMCTOK    1185    A
COMCTOK  *                                                                            COMCTOK    1186    A
COMCTOK  *         E.G. GIVEN THE FOLLOWING --                                        COMCTOK    1187    A
COMCTOK  *                                                                            COMCTOK    1188    A
COMCTOK  *         WORD  0         1          2           3                           COMCTOK    1189    A
COMCTOK  *               ......ALPHA.=.BETA.+.1000........0000000000                  COMCTOK    1190    A
COMCTOK  *                            .        .                                      COMCTOK    1191    A
COMCTOK  *                            .        ......... W2 = 2                       COMCTOK    1192    A
COMCTOK  *                            .                  C2 = 60D                     COMCTOK    1193    A
COMCTOK  *                            .                                               COMCTOK    1194    A
COMCTOK  *                            ................... W1 = 1                       COMCTOK    1195    A
COMCTOK  *                                               C1 = 48D                     COMCTOK    1196    A
COMCTOK  *                                                                            COMCTOK    1197    A
COMCTOK  *         WHERE  W1 = WORD POSITION OF 1ST CHAR.                             COMCTOK    1198    A
COMCTOK  *                C1 = CHAR/BIT POSITION WITHIN *W1* OF 1ST CHAR.             COMCTOK    1199    A
COMCTOK  *                W2 = WORD POSTIONN OF 2ND CHAR.                             COMCTOK    1200    A
COMCTOK  *                C2 = CHAR/BIT POSTIION WITHIN *W2* OF 2ND CHAR.             COMCTOK    1201    A
COMCTOK  *                                                                            COMCTOK    1202    A
COMCTOK  *         NOTE THAT *C1* AND *C2* ARE *TOK* BIT POSITIONS.  THAT IS,         COMCTOK    1203    A
COMCTOK  *         THEY SPECIFY THE NR OF BITS **REMAINING** IN THEIR RESPECTIVE      COMCTOK    1204    A
COMCTOK  *         WORD.  THEREFORE, *C.N* = 60D MEANS LEFT-MOST CHAR IN *W.N*,       COMCTOK    1205    A
COMCTOK  *         *C.N* = 54D MEANS 2ND TO THE LEFT-MOST CHAR IN *W.N*, ETC.         COMCTOK    1206    A
COMCTOK  *                                                                            COMCTOK    1207    A
COMCTOK  *         SPECIAL NOTE ABOUT *C.N* = 0.  IF *C.N* = 0 AND *W.N* = X,         COMCTOK    1208    A
COMCTOK  *         THEN *CCD* TREATS THIS CASE AS EQUIVALENT TO *C.N* = 60D AND       COMCTOK    1209    A
COMCTOK  *         *W.N* = X+1.                                                       COMCTOK    1210    A
COMCTOK  *                                                                            COMCTOK    1211    A
COMCTOK  *         THE FORMULA FOR COMPUTING THE DIFFERENCE BETWEEN (W1,C1) AND       COMCTOK    1212    A
COMCTOK  *         (W2,C2) IS --                                                      COMCTOK    1213    A
COMCTOK  *                                                                            COMCTOK    1214    A
COMCTOK  *                DIFF = ((W2-W1)*(60/CHAR))+(((60-C2)-(60-C1))/CHAR)         COMCTOK    1215    A
COMCTOK  *                                                                            COMCTOK    1216    A
COMCTOK  *         THE CODE TO DO THIS WAS STOLEN FROM THE *FTN* OBJECT CODE          COMCTOK    1217    A
COMCTOK  *         FOR THE ABOVE *FORTRAN* STATEMENT.                                 COMCTOK    1218    A
COMCTOK  *                                                                            COMCTOK    1219    A
COMCTOK  *         ENTRY  (X1) = W1                                                   COMCTOK    1220    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCTOK

```
COMCTOK  *                (X2) = C1                                          COMCTOK   1221   A
COMCTOK  *                (X3) = W2                                          COMCTOK   1222   A
COMCTOK  *                (X4) = C2                                          COMCTOK   1223   A
COMCTOK  *                                                                   COMCTOK   1224   A
COMCTOK  *         EXIT   (X7) = NR OF CHARS BETWEEN (W1,C1) AND (W2,C2).    COMCTOK   1225   A
COMCTOK  *                      NOTE THAT IF (W2,C2) IS TO THE RIGHT OF (W1,C1),  COMCTOK 1226  A
COMCTOK  *                      THEN (X7) IS .PL., WHILE IF (W2,C2) IS TO THE COMCTOK 1227  A
COMCTOK  *                      LEFT OF (W1,C1), THEN (X7) IS .MI.           COMCTOK   1228   A
COMCTOK  *                                                                   COMCTOK   1229   A
COMCTOK  *         USES   X - 1,2,3,4,6,7                                    COMCTOK   1230   A
COMCTOK  *                A - NONE                                           COMCTOK   1231   A
COMCTOK  *                B - 2                                              COMCTOK   1232   A
COMCTOK  *                                                                   COMCTOK   1233   A
COMCTOK  *         CALLS  NONE                                               COMCTOK   1234   A
COMCTOK                                                                      COMCTOK   1235   A
COMCTOK                                                                      COMCTOK   1236   A
COMCTOK   CCD      SUBR                ** ENTRY/EXIT **                      COMCTOK   1237   A
COMCTOK            SX7    60D          (X7) = NR OF BITS IN A WORD           COMCTOK   1238   A
COMCTOK            IX6    X3-X1        (X6) = W2-W1                          COMCTOK   1239   A
COMCTOK            IX2    X7-X2        (X2) = 60-C1                          COMCTOK   1240   A
COMCTOK            IX4    X7-X4        (X4) = 60-C2                          COMCTOK   1241   A
COMCTOK            SX7    CHAR                                               COMCTOK   1242   A
COMCTOK            IX2    X4-X2        (X2) = (60-C2)-(60-C1)                COMCTOK   1243   A
COMCTOK            PX7    X7                                                 COMCTOK   1244   A
COMCTOK            PX2    X2                                                 COMCTOK   1245   A
COMCTOK            SX4    60D/CHAR     (X4) = NR OF CHARS IN A WORD          COMCTOK   1246   A
COMCTOK            NX7    X7                                                 COMCTOK   1247   A
COMCTOK            NX2    X2                                                 COMCTOK   1248   A
COMCTOK            IX4    X6*X4        (X4) = (W2-W1)*(60/CHAR)              COMCTOK   1249   A
COMCTOK            FX2    X2/X7        (X2) = ((60-C2)-(60-C1))/CHAR         COMCTOK   1250   A
COMCTOK            UX3    X2,B2                                              COMCTOK   1251   A
COMCTOK            LX2    X3,B2                                              COMCTOK   1252   A
COMCTOK            IX7    X4+X2                                              COMCTOK   1253   A
COMCTOK            EQ     EXIT.                                              COMCTOK   1254   A
COMCTOK   RER      SPACE  4,10                                               COMCTOK   1255   A
COMCTOK  **        RER - RESTORE ENTOKENING REGISTERS.                       COMCTOK   1256   A
COMCTOK  *                                                                   COMCTOK   1257   A
COMCTOK  *                                                                   COMCTOK   1258   A
COMCTOK  *         ENTRY  ENTOKENING REGISTERS IN *TOKCOM*.                  COMCTOK   1259   A
COMCTOK  *                                                                   COMCTOK   1260   A
COMCTOK  *         EXIT   (B2) = FWA OF *TOKCOM*.                            COMCTOK   1261   A
COMCTOK  *                ENTOKENING REGISTERS SET UP, **EXCEPT** (A6,X6).   COMCTOK   1262   A
COMCTOK  *                (A6,X6) ARE **NOT** RESTORED BECAUSE I COULD FIND  COMCTOK   1263   A
COMCTOK  *                NO GENERAL ALGORITHM/RULE FOR KNOWING WHETHER (A6) COMCTOK   1264   A
COMCTOK  *                SHOULD POINT TO LAST TOKEN GENERATED OR NEXT TOKEN COMCTOK   1265   A
COMCTOK  *                TO GENERATE.  ERGO (A6) IS LEFT ALONE...           COMCTOK   1266   A
COMCTOK  *                                                                   COMCTOK   1267   A
COMCTOK  *         USES   X - 1,2,3,4,5                                      COMCTOK   1268   A
COMCTOK  *                A - 0,1,2,3,4,5                                    COMCTOK   1269   A
COMCTOK  *                B - 2,3,6,7                                        COMCTOK   1270   A
COMCTOK  *                                                                   COMCTOK   1271   A
COMCTOK  *         CALLS  NONE                                               COMCTOK   1272   A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCTOK

```
COMCTOK                                                                                COMCTOK     1273     A
COMCTOK                                                                                COMCTOK     1274     A
COMCTOK   RER      SUBR                ** ENTRY/EXIT **                                COMCTOK     1275     A
COMCTOK            SA1    FW=TCOM                                                      COMCTOK     1276     A
COMCTOK            SB2    X1          (B2) = FWA OF *TOKCOM*                           COMCTOK     1277     A
COMCTOK            =A1    B2+TC.P                                                      COMCTOK     1278     A
COMCTOK            =A2    A1+TC.SOA-TC.P                                               COMCTOK     1279     A
COMCTOK            =A3    A2+TC.SOX-TC.SOA                                             COMCTOK     1280     A
COMCTOK            =A4    A3+TC.SOL-TC.SOX                                             COMCTOK     1281     A
COMCTOK            SA0    X1                                                           COMCTOK     1282     A
COMCTOK            SA5    X2                                                           COMCTOK     1283     A
COMCTOK            BX5    X3                                                           COMCTOK     1284     A
COMCTOK            SB7    X4                                                           COMCTOK     1285     A
COMCTOK                                                                                COMCTOK     1286     A
COMCTOK            =A1    A4+TC.SOB-TC.SOL                                             COMCTOK     1287     A
COMCTOK            =A2    A1+TC.TBB-TC.SOB                                             COMCTOK     1288     A
COMCTOK            =A4    A2+TC.SOC-TC.TBB                                             COMCTOK     1289     A
COMCTOK            SB6    X1                                                           COMCTOK     1290     A
COMCTOK            SB3    X2                                                           COMCTOK     1291     A
COMCTOK            EQ     EXIT.                                                        COMCTOK     1292     A
COMCTOK   SER      SPACE  4,10                                                         COMCTOK     1293     A
COMCTOK  **        SER - SAVE ENTOKENING REGISTERS.                                    COMCTOK     1294     A
COMCTOK  *                                                                             COMCTOK     1295     A
COMCTOK  *                                                                             COMCTOK     1296     A
COMCTOK  *         ENTRY  ENTOKENING REGISTERS INTACT.                                 COMCTOK     1297     A
COMCTOK  *                                                                             COMCTOK     1298     A
COMCTOK  *         EXIT   (B2) = FWA OF *TOKCOM*.                                      COMCTOK     1299     A
COMCTOK  *                ENTOKENING REGISTERS STILL INTACT AND SAVED                  COMCTOK     1300     A
COMCTOK  *                IN *TOKCOM*.                                                 COMCTOK     1301     A
COMCTOK  *                                                                             COMCTOK     1302     A
COMCTOK  *         USES   X - 1,7                                                      COMCTOK     1303     A
COMCTOK  *                A - 1,7                                                      COMCTOK     1304     A
COMCTOK  *                B - 2                                                        COMCTOK     1305     A
COMCTOK  *                                                                             COMCTOK     1306     A
COMCTOK  *         CALLS  NONE                                                         COMCTOK     1307     A
COMCTOK                                                                                COMCTOK     1308     A
COMCTOK                                                                                COMCTOK     1309     A
COMCTOK   SER      SUBR                ** ENTRY/EXIT **                                COMCTOK     1310     A
COMCTOK            SA1    FW=TCOM                                                      COMCTOK     1311     A
COMCTOK            SB2    X1          (B2) = FWA OF *TOKCOM*                           COMCTOK     1312     A
COMCTOK            SX7    A0-B0                                                        COMCTOK     1313     A
COMCTOK            =A7    B2+TC.P                                                      COMCTOK     1314     A
COMCTOK            SX7    A5-B0                                                        COMCTOK     1315     A
COMCTOK            =A7    A7+TC.SOA-TC.P                                               COMCTOK     1316     A
COMCTOK            BX7    X5                                                           COMCTOK     1317     A
COMCTOK            =A7    A7+TC.SOX-TC.SOA                                             COMCTOK     1318     A
COMCTOK            SX7    B7-B0                                                        COMCTOK     1319     A
COMCTOK            =A7    A7+TC.SOL-TC.SOX                                             COMCTOK     1320     A
COMCTOK            SX7    B6-B0                                                        COMCTOK     1321     A
COMCTOK            =A7    A7+TC.SOB-TC.SOL                                             COMCTOK     1322     A
COMCTOK            BX7    X4                                                           COMCTOK     1323     A
COMCTOK            =A7    A7+TC.SOC-TC.SOB                                             COMCTOK     1324     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMCTOK

```
COMCTOK            SX7    A6-B0                                                    COMCTOK    1325    A
COMCTOK            =A7    A7+TC.TBA-TC.SOC                                         COMCTOK    1326    A
COMCTOK            SX7    B3-B0                                                    COMCTOK    1327    A
COMCTOK            =A7    A7+TC.TBB-TC.TBA                                         COMCTOK    1328    A
COMCTOK            EQ     EXIT.                                                    COMCTOK    1329    A
COMCTOK   TOK      SPACE  4,10                                                     COMCTOK    1330    A
COMCTOK   QUAL$    IF     -DEF,QUAL$                                               COMCTOK    1331    A
COMCTOK            QUAL   *                                                        COMCTOK    1332    A
COMCTOK   TOK      =      /COMCTOK/TOK                                             COMCTOK    1333    A
COMCTOK   CCD      =      /COMCTOK/CCD                                             COMCTOK    1334    A
COMCTOK   RER      =      /COMCTOK/RER                                             COMCTOK    1335    A
COMCTOK   SER      =      /COMCTOK/SER                                             COMCTOK    1336    A
COMCTOK   TF=SQZ   =      /COMCTOK/TF=SQZ                                          COMCTOK    1337    A
COMCTOK                                                                           COMCTOK    1338    A
COMCTOK            IFNE   TEST,0,1                                                 COMCTOK    1339    A
COMCTOK   TF=IDP   =      /COMCTOK/TF=IDP                                          COMCTOK    1340    A
COMCTOK   TOK=MN   =      /COMCTOK/TOK=MN                                          COMCTOK    1341    A
COMCTOK   TOK=COS  =      /COMCTOK/TOK=COS                                         COMCTOK    1342    A
COMCTOK   TOK=CON  =      /COMCTOK/TOK=CON                                         COMCTOK    1343    A
COMCTOK   QUAL$    ENDIF                                                          COMCTOK    1344    A
COMCTOK   TOK      SPACE  4,10                                                     COMCTOK    1345    A
COMCTOK   COMCTOK  ENDX                                                           COMCTOK    1346    A
```

SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - COMCTOK

IDENTIFIER	TOTAL	ACTIVE
COMCTOK	1346	1346

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMDTOK

```
COMDTOK  *COMDECK COMDTOK                                                         COMDTOK     1    A
COMDTOK   COMDTOK  CTEXT  COMDTOK 2.0 - *TEST* MODE DEBUG UTILITY FOR *COMCTOK*.  COMDTOK     2    A
COMDTOK   TOK      SPACE  4                                                       COMDTOK     3    A
COMDTOK            IF     -DEF,QUAL$,1                                            COMDTOK     4    A
COMDTOK            QUAL   COMDTOK                                                 COMDTOK     5    A
COMDTOK            BASE   D                                                       COMDTOK     6    A
COMDTOK   TOK      SPACE  4,10                                                    COMDTOK     7    A
COMDTOK  ***       COMDTOK - *TEST* MODE DEBUG UTILITY FOR *COMCTOK*.             COMDTOK     8    A
COMDTOK  *         P CLARE   77/02/26                                             COMDTOK     9    A
COMDTOK  *         P CLARE   78/04/27 VERSION 2.0                                 COMDTOK    10    A
COMDTOK   TOK      SPACE  4,10                                                    COMDTOK    11    A
COMDTOK  ***       COMDTOK - *TEST* MODE DEBUG UTILITY FOR *COMCTOK*.             COMDTOK    12    A
COMDTOK  *                                                                        COMDTOK    13    A
COMDTOK  *                                                                        COMDTOK    14    A
COMDTOK  *         THIS DECK CONTAINS A FEW *TEST* MODE DEBUGGING UTILITY         COMDTOK    15    A
COMDTOK  *         SUBROUTINES THAT CAN BE USEFUL IN HELPING ONE DEBUG THEIR      COMDTOK    16    A
COMDTOK  *         *TOGEL* PROGRAMS.                                              COMDTOK    17    A
COMDTOK  *                                                                        COMDTOK    18    A
COMDTOK  *         *COMDTOK* CAN BE USED TO DEASSEMBLE *TOGEL* BINARY             COMDTOK    19    A
COMDTOK  *         INSTRUCTIONS (ASSEMBLED VIA *COMATOK*), AND TO PROVIDE         COMDTOK    20    A
COMDTOK  *         THE ABILITY TO PLACE *COMCTOK* IN STEP MODE.  THAT IS,         COMDTOK    21    A
```

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMDTOK

```
COMDTOK  *         TO ALLOW THE PROGRAMMER TO CONTROL THE EXECUTION/INTERPRETING          COMDTOK      22     A
COMDTOK  *         OF *TOGEL* BINARY INSTRUCTIONS BY *COMCTOK*, AND TO ALLOW              COMDTOK      23     A
COMDTOK  *         THE PROGRAMMER TO OBSERVE THE GENERATION OF TOKENS                     COMDTOK      24     A
COMDTOK  *         BY *COMCTOK* ON AN INSTRUCTION BY INSTRUCTION BASIS.                   COMDTOK      25     A
COMDTOK  *                                                                                COMDTOK      26     A
COMDTOK  *         IN THE FOLLOWING PARAGRAPHS, *ASSEMBLY-TIME DEPENDENCIES*              COMDTOK      27     A
COMDTOK  *         REFERS TO INTERFACES THAT THE *COMDTOK* INSTALLER MUST                 COMDTOK      28     A
COMDTOK  *         PROVIDE.  *ASSEMBLY-TIME OPTIONS* REFERS TO INTERFACES                 COMDTOK      29     A
COMDTOK  *         THAT ARE OPTIONAL, I.E. THAT HAVE A DEFAULT VALUE.                     COMDTOK      30     A
COMDTOK  *                                                                                COMDTOK      31     A
COMDTOK  ************************************************************************        COMDTOK      32     A
COMDTOK  *                                                                                COMDTOK      33     A
COMDTOK  *         ASSEMBLY-TIME DEPENDENCIES --                                          COMDTOK      34     A
COMDTOK  *                                                                                COMDTOK      35     A
COMDTOK  ************************************************************************        COMDTOK      36     A
COMDTOK  *                                                                                COMDTOK      37     A
COMDTOK  *         *COMCTOK* (TOKEN GENERATOR) MUST BE ASSEMBLED WITH THE                 COMDTOK      38     A
COMDTOK  *         SYMBOL *TEST* SET TO .NZ.  THIS WILL FORCE ASSEMBLY OF                 COMDTOK      39     A
COMDTOK  *         CODE IN *COMCTOK* TO INTERFACE WITH *COMDTOK*.                         COMDTOK      40     A
COMDTOK  *                                                                                COMDTOK      41     A
COMDTOK  ************************************************************************        COMDTOK      42     A
COMDTOK  *                                                                                COMDTOK      43     A
COMDTOK  *         *COMSTOK*, WHICH CONTAINS THE SYMBOLS THAT DESCRIBE/DEFINE             COMDTOK      44     A
COMDTOK  *         THE *TOGEL* BINARY INSTRUCTION FORMATS, MUST BE PROVIDED.              COMDTOK      45     A
COMDTOK  *                                                                                COMDTOK      46     A
COMDTOK  ************************************************************************        COMDTOK      47     A
COMDTOK  *                                                                                COMDTOK      48     A
COMDTOK  *         *COMDTOK* INTERFACES WITH *COMCIDP* (FORMERLY *DBG=IDP*),              COMDTOK      49     A
COMDTOK  *         THE INTERACTIVE DEBUG PACKAGE, AND THEREFORE REQUIRES                  COMDTOK      50     A
COMDTOK  *         *COMSIDP*, WHICH CONTAINS *IDP* INTERFACE MACRO AND SYMBOL             COMDTOK      51     A
COMDTOK  *         DEFINITIONS.                                                           COMDTOK      52     A
COMDTOK  *                                                                                COMDTOK      53     A
COMDTOK  ************************************************************************        COMDTOK      54     A
COMDTOK  *                                                                                COMDTOK      55     A
COMDTOK  *         SUMMARY OF REQUIRED COMDECKS --                                        COMDTOK      56     A
COMDTOK  *                                                                                COMDTOK      57     A
COMDTOK  *         COMCIDP   COMCMCS   COMCTOK                                            COMDTOK      58     A
COMDTOK  *                                                                                COMDTOK      59     A
COMDTOK  *         COMSIDP   COMSTOK                                                      COMDTOK      60     A
COMDTOK  *                                                                                COMDTOK      61     A
COMDTOK  *         NOTE-- *COMCIDP* AND *COMCTOK* HAVE THEIR OWN SET OF REQUIRED          COMDTOK      62     A
COMDTOK  *         COMDECKS.  THIS LIST ASSUMES THAT THE PROGRAMMER HAS ALREADY           COMDTOK      63     A
COMDTOK  *         CONFIGURED AND INSTALLED *COMCIDP* AND *COMCTOK*.                      COMDTOK      64     A
COMDTOK  *                                                                                COMDTOK      65     A
COMDTOK  ************************************************************************        COMDTOK      66     A
COMDTOK   TOGELI   SPACE  4,10                                                            COMDTOK      67     A
COMDTOK  **        TOGELI - DEFINE *TOGEL* INSTRUCTION/OPCODE.                            COMDTOK      68     A
COMDTOK  *                                                                                COMDTOK      69     A
COMDTOK  *                                                                                COMDTOK      70     A
COMDTOK  *         THIS MACRO DEFINES A TABLE ENTRY THAT CAN BE USED                      COMDTOK      71     A
COMDTOK  *         BY EITHER *TID* OR *TOD* IN PROCESSING A BINARY *TOGEL*                COMDTOK      72     A
COMDTOK  *         INSTRUCTION.  SEE *TID/TOD*.                                           COMDTOK      73     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMDTOK

```
COMDTOK  *                                                                COMDTOK     74   A
COMDTOK  * TOFU    TOGELI DPC                                             COMDTOK     75   A
COMDTOK  *                                                                COMDTOK     76   A
COMDTOK  * COF     TOGELI CASEOF                                          COMDTOK     77   A
COMDTOK  *                                                                COMDTOK     78   A
COMDTOK  *         ENTRY  TOFU = TOFU SUFFIX FOR THIS *TOGEL* INSTRUCTION,  COMDTOK  79   A
COMDTOK  *                      I.E. PROCESSOR ADDRESSES ARE FORMED BY SUFFIXING  COMDTOK  80   A
COMDTOK  *                      *TOFU* TO *TOK=*, *TOD=*, AND *TID=*.     COMDTOK     81   A
COMDTOK  *                 DPC  = DISPLAY CODE OPCODE FOR THIS *TOGEL* BINARY  COMDTOK  82   A
COMDTOK  *                      INSTRUCTION.                              COMDTOK     83   A
COMDTOK  *                                                                COMDTOK     84   A
COMDTOK  *         EXIT   NONE                                            COMDTOK     85   A
COMDTOK  *                                                                COMDTOK     86   A
COMDTOK  *         USES   NONE                                            COMDTOK     87   A
COMDTOK  *                                                                COMDTOK     88   A
COMDTOK  *         CALLS  NONE                                            COMDTOK     89   A
COMDTOK                                                                   COMDTOK     90   A
COMDTOK                                                                   COMDTOK     91   A
COMDTOK            PURGMAC TOGELI                                         COMDTOK     92   A
COMDTOK                                                                   COMDTOK     93   A
COMDTOK            MACRO  TOGELI,TOFU,DPC                                 COMDTOK     94   A
COMDTOK  *                                                                COMDTOK     95   A
COMDTOK   +        VFD    6/0,18/TID=TOFU,18/TOD=TOFU,18/=XTOK=TOFU       COMDTOK     96   A
COMDTOK  *                                                                COMDTOK     97   A
COMDTOK   +        DATA   L/DPC/                                          COMDTOK     98   A
COMDTOK  *                                                                COMDTOK     99   A
COMDTOK   TOGELI   ENDM                                                   COMDTOK    100   A
COMDTOK   FW.OPC   SPACE  4,10                                            COMDTOK    101   A
COMDTOK  **        FW.OPC - *TOGEL* INSTRUCTION OPCODE TABLE.             COMDTOK    102   A
COMDTOK                                                                   COMDTOK    103   A
COMDTOK                                                                   COMDTOK    104   A
COMDTOK   FW.OPC   BSS    0          ** FWA OF OPCODE TABLE **            COMDTOK    105   A
COMDTOK   CAL      TOGELI CALT                                            COMDTOK    106   A
COMDTOK   COF      TOGELI CASEOF                                          COMDTOK    107   A
COMDTOK   ELS      TOGELI ELST                                            COMDTOK    108   A
COMDTOK   ENT      TOGELI ENDT                                            COMDTOK    109   A
COMDTOK   GN       TOGELI GROUP                                           COMDTOK    110   A
COMDTOK   GO       TOGELI GOTO                                            COMDTOK    111   A
COMDTOK   GS       TOGELI GROUP                                           COMDTOK    112   A
COMDTOK   IFT      TOGELI IFT                                             COMDTOK    113   A
COMDTOK   ON       TOGELI ON                                              COMDTOK    114   A
COMDTOK   XEQ      TOGELI XEQ                                             COMDTOK    115   A
COMDTOK                                                                   COMDTOK    116   A
COMDTOK   L.OPC    =      *-FW.OPC                                        COMDTOK    117   A
COMDTOK   PTR.     SPACE  4,10                                            COMDTOK    118   A
COMDTOK  **        PTR. - DEFINE *PTR* (POINTER) NRS FOR *COMDTOK*.       COMDTOK    119   A
COMDTOK                                                                   COMDTOK    120   A
COMDTOK                                                                   COMDTOK    121   A
COMDTOK   PTR.TOG  =      4                                               COMDTOK    122   A
COMDTOK   PTR.TOK  =      5                                               COMDTOK    123   A
COMDTOK   TOG=     SPACE  4,10                                            COMDTOK    124   A
COMDTOK  **        TOG= - SAVED FWA AND LWA FOR *TOGEL*.                  COMDTOK    125   A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMDTOK

```
COMDTOK                                                                  COMDTOK      126     A
COMDTOK                                                                  COMDTOK      127     A
COMDTOK   TOG=FWA  BSSZ   2                                              COMDTOK      128     A
COMDTOK   TOG=LWA  =      TOG=FWA+1                                      COMDTOK      129     A
COMDTOK   UKT=TOK  SPACE  4,10                                           COMDTOK      130     A
COMDTOK  **        UKT=TOK - USER IDP KEYWORD TABLE FOR *COMDTOK*.       COMDTOK      131     A
COMDTOK                                                                  COMDTOK      132     A
COMDTOK                                                                  COMDTOK      133     A
COMDTOK   UKT=TOK  BSS    0           ** FWA OF KEYWORD TABLE **         COMDTOK      134     A
COMDTOK   BRT      KEYW   (BRTOK TOGEL)                                  COMDTOK      135     A
COMDTOK   TOG      KEYW   (TOGEL FWA,LWA,LEN)                            COMDTOK      136     A
COMDTOK   TOK      KEYW   (TOKEN FWA,LWA,LEN)                            COMDTOK      137     A
COMDTOK   UBT      KEYW   (UBTOK)                                        COMDTOK      138     A
COMDTOK            DATA   0           ** END OF TABLE **                 COMDTOK      139     A
COMDTOK   UKT=LNK  DATA   0           ** LINK WORD **                    COMDTOK      140     A
COMDTOK   ST=BRT   SPACE  4,10                                           COMDTOK      141     A
COMDTOK  **        HERE TO PROCESS *BRTOK TOGEL*.                        COMDTOK      142     A
COMDTOK                                                                  COMDTOK      143     A
COMDTOK   ST=BRT   BSS    0           ** ENTRY **                        COMDTOK      144     A
COMDTOK            NZ     X5,BRT1     IF KEYWORD NOT IMMEDIATELY FOLLOWED BY EOS  COMDTOK  145  A
COMDTOK            SX6    -1                                              COMDTOK      146     A
COMDTOK            SA6    =XTF=IDP    SET TO *BREAK ON ANY TOGEL INSTRUCTION*  COMDTOK  147  A
COMDTOK            EQ     IDP=MN                                          COMDTOK      148     A
COMDTOK                                                                  COMDTOK      149     A
COMDTOK  *         SEARCH FOR *TOGEL* DPC OPCODE.                        COMDTOK      150     A
COMDTOK                                                                  COMDTOK      151     A
COMDTOK   BRT1     SB7    L.OPC       (B7) = LEN OF *TOGEL* OPCODE TABLE COMDTOK      152     A
COMDTOK            MX1    TB.TOCL                                         COMDTOK      153     A
COMDTOK            SA5    A5+B1       (X5) = TOKEN CONTAINING *TOGEL* OPC COMDTOK      154     A
COMDTOK  *                                   TO BREAK AT                 COMDTOK      155     A
COMDTOK            SX6    X5-O.VAR                                        COMDTOK      156     A
COMDTOK            NZ     X6,IDP=ER   IF NOT *VAR* TOKEN, ERROR...        COMDTOK      157     A
COMDTOK            BX6    X1*X5       (X6) = TOGEL INSTRUCTION MNEMONIC, -L- FMT  COMDTOK  158  A
COMDTOK            SA1    FW.OPC+1    (A1,X1) = A+C OF 1ST DPC MNEM IN OPC TBL  COMDTOK  159  A
COMDTOK                                                                  COMDTOK      160     A
COMDTOK   BRT2     LE     B7,B0,IDP=ER  IF EOT AND NO FIND, ERROR...     COMDTOK      161     A
COMDTOK            SB2    A1          (B2) = ADDR+1 OF FOUND ENTRY, IF FOUND  COMDTOK  162  A
COMDTOK            BX7    X1-X6                                           COMDTOK      163     A
COMDTOK            SA1    A1+2                                            COMDTOK      164     A
COMDTOK            SB7    B7-2                                            COMDTOK      165     A
COMDTOK            NZ     X7,BRT2     IF NO FIND                          COMDTOK      166     A
COMDTOK                                                                  COMDTOK      167     A
COMDTOK  *         HERE WITH A FIND.  SET UP (TF=IDP).                   COMDTOK      168     A
COMDTOK                                                                  COMDTOK      169     A
COMDTOK            SA2    B2-B1       (X2) = OPC TBL ENTRY CONTAINING TOFU ADDR  COMDTOK  170  A
COMDTOK            SX6    X2          (X6) = TOFU ADDR                    COMDTOK      171     A
COMDTOK            SA6    TF=IDP                                          COMDTOK      172     A
COMDTOK            EQ     IDP=MN                                          COMDTOK      173     A
COMDTOK   ST=TOG   SPACE  4,10                                           COMDTOK      174     A
COMDTOK  **        HERE TO PROCESS *TOGEL FWA,LWA,LEN*.                  COMDTOK      175     A
COMDTOK                                                                  COMDTOK      176     A
COMDTOK   ST=TOG   BSS    0           ** ENTRY **                        COMDTOK      177     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMDTOK

```
COMDTOK            RJ     UTG         USER *TOGEL* OWNCODE                          COMDTOK      178     A
COMDTOK            RJ     =XPAT       PARSE *FWA,LWA,LEN* TRIPLE                    COMDTOK      179     A
COMDTOK            MI     B2,IDP=ER   IF FWA, LWA, OR LEN IS BAD                    COMDTOK      180     A
COMDTOK            BX6    X2                                                        COMDTOK      181     A
COMDTOK            LX7    X3                                                        COMDTOK      182     A
COMDTOK            SA6    TOG=FWA                                                   COMDTOK      183     A
COMDTOK            SA7    TOG=LWA                                                   COMDTOK      184     A
COMDTOK                                                                             COMDTOK      185     A
COMDTOK   ST=TOG2  SA1    TOG=FWA                                                   COMDTOK      186     A
COMDTOK            SA2    A1+B1                                                     COMDTOK      187     A
COMDTOK            IX3    X2-X1                                                     COMDTOK      188     A
COMDTOK            SX6    X1+B1                                                     COMDTOK      189     A
COMDTOK            MI     X3,IDP=MN   IF FINISHED                                   COMDTOK      190     A
COMDTOK            SA6    A1                                                        COMDTOK      191     A
COMDTOK            RJ     TOD         TOGEL DEASSEMBLER                             COMDTOK      192     A
COMDTOK            MI     B2,*+4S15   IF SOME ADDR WAS BAD                          COMDTOK      193     A
COMDTOK            EQ     ST=TOG2                                                   COMDTOK      194     A
COMDTOK   ST=TOK   SPACE  4,10                                                      COMDTOK      195     A
COMDTOK  **        HERE TO PROCESS *TOKEN FWA,LWA,LEN*.                             COMDTOK      196     A
COMDTOK                                                                             COMDTOK      197     A
COMDTOK   ST=TOK   BSS    0           ** ENTRY **                                   COMDTOK      198     A
COMDTOK            RJ     UTK         USER *TOKEN* OWNCODE                          COMDTOK      199     A
COMDTOK            RJ     PAT         PARSE *FWA,LWA,LEN* TRIPLE                    COMDTOK      200     A
COMDTOK            MI     B2,IDP=ER   IF FWA, LWA, OR LEN IS BAD                    COMDTOK      201     A
COMDTOK            RJ     LTK         LIST TOKENS                                   COMDTOK      202     A
COMDTOK            EQ     IDP=MN                                                    COMDTOK      203     A
COMDTOK   ST=UBT   SPACE  4,10                                                      COMDTOK      204     A
COMDTOK  **        HERE TO PROCESS *UBTOK*.                                         COMDTOK      205     A
COMDTOK                                                                             COMDTOK      206     A
COMDTOK   ST=UBT   BSS    0           ** ENTRY **                                   COMDTOK      207     A
COMDTOK            SX6    0                                                         COMDTOK      208     A
COMDTOK            SA6    TF=IDP      SET TO *NO BRTOK SELECTED*                    COMDTOK      209     A
COMDTOK            EQ     IDP=MN                                                    COMDTOK      210     A
COMDTOK   CTD      SPACE  4,10                                                      COMDTOK      211     A
COMDTOK  **        CTD - CONVERT TOKEN TYPE TO DPC.                                 COMDTOK      212     A
COMDTOK  *                                                                          COMDTOK      213     A
COMDTOK  *                                                                          COMDTOK      214     A
COMDTOK  *         THIS ROUTINE CONVERTS A BINARY TOKEN TYPE VALUE TO DPC           COMDTOK      215     A
COMDTOK  *         BY 1 OF 2 METHODS --                                             COMDTOK      216     A
COMDTOK  *                                                                          COMDTOK      217     A
COMDTOK  *           1. IF THE USER PROVIDED *CTD=*, THEN CALL *CTD=*               COMDTOK      218     A
COMDTOK  *              TO PERFORM THE CONVERSION, ELSE                             COMDTOK      219     A
COMDTOK  *                                                                          COMDTOK      220     A
COMDTOK  *           2. IF THE USER DID NOT PROVIDE *CTD=*, THEN CALL *COD*         COMDTOK      221     A
COMDTOK  *              TO CONVERT THE TOKEN TYPE TO OCTAL DPC.                     COMDTOK      222     A
COMDTOK  *                                                                          COMDTOK      223     A
COMDTOK  *         ENTRY  (X1) = TOKEN TYPE, IN BINARY.                             COMDTOK      224     A
COMDTOK  *                                                                          COMDTOK      225     A
COMDTOK  *         EXIT   (X6) = TOKEN TYPE, IN -L- FMT.                            COMDTOK      226     A
COMDTOK  *                (X1) = (X6)                                               COMDTOK      227     A
COMDTOK  *                (B2) = .PL. IF USER PROVIDED *CTD=*, ELSE .MI.            COMDTOK      228     A
COMDTOK  *                                                                          COMDTOK      229     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMDTOK

```
COMDTOK  *         USES   ALL BUT A0,X0,A5,X5,B7                                COMDTOK   230   A
COMDTOK  *                                                                      COMDTOK   231   A
COMDTOK  *         CALLS  COD,CTD(IF AVAILABLE)                                 COMDTOK   232   A
COMDTOK                                                                         COMDTOK   233   A
COMDTOK                                                                         COMDTOK   234   A
COMDTOK   CTD      SUBR                ** ENTRY/EXIT **                         COMDTOK   235   A
COMDTOK            SX6    B7                                                    COMDTOK   236   A
COMDTOK            SB2    =YCTD=                                                COMDTOK   237   A
COMDTOK            SA6    CTDA        SAVE (B7)                                 COMDTOK   238   A
COMDTOK            MI     B2,CTD2     IF USER DID NOT PROVIDE *CTD=*            COMDTOK   239   A
COMDTOK            RJ     =YCTD=      CONVERT TOKEN TYPE TO DPC                 COMDTOK   240   A
COMDTOK            EQ     CTD3                                                  COMDTOK   241   A
COMDTOK                                                                         COMDTOK   242   A
COMDTOK   CTD2     RJ     COD         CONVERT BINARY TO OCTAL DPC               COMDTOK   243   A
COMDTOK            MX2    1                                                     COMDTOK   244   A
COMDTOK            SB2    B2-B1       (B2) = NR OF BITS CONVERTED - 1           COMDTOK   245   A
COMDTOK            AX2    B2          (X2) = MASK FOR EXTRACTING CONVERTED TOT  COMDTOK   246   A
COMDTOK            BX6    X2*X4       (X6) = TOKEN TYPE, IN -L- FMT             COMDTOK   247   A
COMDTOK                                                                         COMDTOK   248   A
COMDTOK   CTD3     SA2    CTDA                                                  COMDTOK   249   A
COMDTOK            SB2    =YCTD=                                                COMDTOK   250   A
COMDTOK            BX1    X6                                                    COMDTOK   251   A
COMDTOK            SB7    X2          RESTORE (B7)                              COMDTOK   252   A
COMDTOK            EQ     EXIT.                                                 COMDTOK   253   A
COMDTOK                                                                         COMDTOK   254   A
COMDTOK                                                                         COMDTOK   255   A
COMDTOK   CTDA     BSSZ   1           SAVED (B7)                               COMDTOK   256   A
COMDTOK   DCR      SPACE  4,10                                                 COMDTOK   257   A
COMDTOK  **        DCR - DEASSEMBLE CHARACTER SHIFT MASK.                       COMDTOK   258   A
COMDTOK  *                                                                      COMDTOK   259   A
COMDTOK  *                                                                      COMDTOK   260   A
COMDTOK  *         THIS ROUTINE DEASSEMBLES A CHARACTER RANGE SHIFT MASK        COMDTOK   261   A
COMDTOK  *         AS WOULD OCCUR IN A *GROUP* OR *IFT* TOGEL BINARY            COMDTOK   262   A
COMDTOK  *         INSTRUCTION, AND FORMATS IT FOR PRINTING.                    COMDTOK   263   A
COMDTOK  *                                                                      COMDTOK   264   A
COMDTOK  *         ENTRY  (X4) = CHAR RANGE SHIFT MASK                          COMDTOK   265   A
COMDTOK  *                (B7) = ADDR OF CURRENT WORD IN PRINT LINE IMAGE       COMDTOK   266   A
COMDTOK  *                                                                      COMDTOK   267   A
COMDTOK  *         EXIT   (B7) = ADDR OF CURRENT WORD IN PRINT LINE IMAGE,      COMDTOK   268   A
COMDTOK  *                       UPDATED IF APPROPRIATE.                        COMDTOK   269   A
COMDTOK  *                                                                      COMDTOK   270   A
COMDTOK  *         USES   X - ALL BUT X0,X5                                     COMDTOK   271   A
COMDTOK  *                A - 2,3,6,7                                           COMDTOK   272   A
COMDTOK  *                B - 1,2,4,5,7                                         COMDTOK   273   A
COMDTOK  *                                                                      COMDTOK   274   A
COMDTOK  *         CALLS  MCS                                                   COMDTOK   275   A
COMDTOK                                                                         COMDTOK   276   A
COMDTOK                                                                         COMDTOK   277   A
COMDTOK   DCR      SUBR                ** ENTRY/EXIT **                         COMDTOK   278   A
COMDTOK                                                                         COMDTOK   279   A
COMDTOK  *         PUT OUT LEFT PARENTHESIS DELIMITER.                          COMDTOK   280   A
COMDTOK                                                                         COMDTOK   281   A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMDTOK

```
COMDTOK            SX1    1R(                                                  COMDTOK     282    A
COMDTOK            LX1    10*CHAR-1*CHAR                                       COMDTOK     283    A
COMDTOK            RJ     =XMCS       MERGE CODED STRING                       COMDTOK     284    A
COMDTOK                                                                        COMDTOK     285    A
COMDTOK  *         FIGURE OUT WHETHER TO USE A SPECIAL SYNTAX.                 COMDTOK     286    A
COMDTOK  *         SPECIAL SYNTAXS ARE --                                      COMDTOK     287    A
COMDTOK  *           (..)   IF ALL BITS ON, (X4) = -0                          COMDTOK     288    A
COMDTOK  *           (-..)  IF ALL BITS OFF, (X4) = +0                         COMDTOK     289    A
COMDTOK  *           (-XXX) WHERE *XXX* IS THE LOGICAL COMPLEMENT OF (X4),     COMDTOK     290    A
COMDTOK  *                  IF MORE BITS ARE ON THAN OFF.                      COMDTOK     291    A
COMDTOK                                                                        COMDTOK     292    A
COMDTOK            CX2    X4          (X2) = NR OF *ON* BITS IN (X4)           COMDTOK     293    A
COMDTOK            SX3    X2-30                                                COMDTOK     294    A
COMDTOK            MI     X3,DCR2     IF NOT USING *-* SYNTAX                  COMDTOK     295    A
COMDTOK            SX1    1R-                                                  COMDTOK     296    A
COMDTOK            BX4    -X4                                                  COMDTOK     297    A
COMDTOK            LX1    10*CHAR-1*CHAR                                       COMDTOK     298    A
COMDTOK            RJ     =XMCS       MERGE CODED STRINGS                      COMDTOK     299    A
COMDTOK                                                                        COMDTOK     300    A
COMDTOK   DCR2     NZ     X4,DCR2A    IF NOT USING *..* SYNTAX                 COMDTOK     301    A
COMDTOK            SX1    2R..                                                 COMDTOK     302    A
COMDTOK            LX1    10*CHAR-2*CHAR                                       COMDTOK     303    A
COMDTOK            RJ     =XMCS       MERGE CODED STRINGS                      COMDTOK     304    A
COMDTOK            EQ     DCR4                                                 COMDTOK     305    A
COMDTOK                                                                        COMDTOK     306    A
COMDTOK  *         INITIALIZE FOR MAIN LOOP.  SET UP (X4) SO THAT WE BEGIN     COMDTOK     307    A
COMDTOK  *         AT 1ST CHAR POSITION IN DISPLAY CODE, NOT ROTATED CHAR      COMDTOK     308    A
COMDTOK  *         SET REPRESENTATION.                                         COMDTOK     309    A
COMDTOK                                                                        COMDTOK     310    A
COMDTOK   DCR2A    SB4    00B         (B4) = CURRENT CHAR IN DPC -R- FMT       COMDTOK     311    A
COMDTOK            SB5    60D         (B5) = NR OF BITS IN A WORD              COMDTOK     312    A
COMDTOK            LX4    -BICH                                                COMDTOK     313    A
COMDTOK            PL     X4,DCR3     IF : NOT SELECTED                        COMDTOK     314    A
COMDTOK            SX1    2R                                                   COMDTOK     315    A
COMDTOK            LX1    10*CHAR-2*CHAR                                       COMDTOK     316    A
COMDTOK            RJ     =XMCS       MERGE CODED STRINGS                      COMDTOK     317    A
COMDTOK                                                                        COMDTOK     318    A
COMDTOK  *         ** MAIN LOOP NODE **                                        COMDTOK     319    A
COMDTOK                                                                        COMDTOK     320    A
COMDTOK   DCR3     LX4    1                                                    COMDTOK     321    A
COMDTOK            SB4    B4+B1                                                COMDTOK     322    A
COMDTOK            GE     B4,B5,DCR4  IF DONE                                  COMDTOK     323    A
COMDTOK            PL     X4,DCR3     IF BIT IS *OFF*                          COMDTOK     324    A
COMDTOK                                                                        COMDTOK     325    A
COMDTOK  *         HERE IF BITS IS *ON*, OUTPUT CHARACTER.                     COMDTOK     326    A
COMDTOK                                                                        COMDTOK     327    A
COMDTOK            SX1    B4                                                   COMDTOK     328    A
COMDTOK            LX1    10*CHAR-1*CHAR                                       COMDTOK     329    A
COMDTOK            RJ     =XMCS       MERGE CODED STRINGS                      COMDTOK     330    A
COMDTOK            EQ     DCR3                                                 COMDTOK     331    A
COMDTOK                                                                        COMDTOK     332    A
COMDTOK  *         HERE WHEN DONE.  PUT OUT TERMINATING DELIMITING RIGHT PAREN. COMDTOK     333    A
```

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMDTOK

```
COMDTOK                                                                  COMDTOK     334    A
COMDTOK   DCR4     SX1    1R)                                            COMDTOK     335    A
COMDTOK            LX1    10*CHAR-1*CHAR                                 COMDTOK     336    A
COMDTOK            RJ     =XMCS       MERGE CODED STRINGS                COMDTOK     337    A
COMDTOK            EQ     EXIT.                                          COMDTOK     338    A
COMDTOK   LTK      SPACE  4,10                                           COMDTOK     339    A
COMDTOK  **        LTK - LIST TOKENS.                                    COMDTOK     340    A
COMDTOK  *                                                               COMDTOK     341    A
COMDTOK  *                                                               COMDTOK     342    A
COMDTOK  *         THIS ROUTINE LISTS TOKENS IN THE FOLLOWING FORMAT --  COMDTOK     343    A
COMDTOK  *                                                               COMDTOK     344    A
COMDTOK  *     COL 1         1         1         1         1         1         1   COMDTOK     345    A
COMDTOK  *         .NNNNNN.+.CCCCCCCCCCCCCCCCCCCC.....DDDDDDDDDD.....TTTTTTTTTT   COMDTOK     346    A
COMDTOK  *                                                               COMDTOK     347    A
COMDTOK  *         WHERE  . = BLANK(55B)                                 COMDTOK     348    A
COMDTOK  *                N = ADDR OF TOKEN  (EITHER RELATIVE OR ABSOLUTE)  COMDTOK     349    A
COMDTOK  *                + = + IF *NNNNNN* IS DECK RELATIVE, ELSE       COMDTOK     350    A
COMDTOK  *                  = BLANK(55B) IF *NNNNNN* IS ABSOLUTE.        COMDTOK     351    A
COMDTOK  *                C = CONVERTED TOKEN (20 OCTAL DIGITS).         COMDTOK     352    A
COMDTOK  *                D = UNCONVERTED TOKEN (ALL 00B CHARS CONVERTED COMDTOK     353    A
COMDTOK  *                    TO BLANKS)                                 COMDTOK     354    A
COMDTOK  *                T = *TMAP* CHARS FOR THIS TOKEN IF *FW.TMAP* DEFINED,  COMDTOK     355    A
COMDTOK  *                    ELSE NULL.                                 COMDTOK     356    A
COMDTOK  *                                                               COMDTOK     357    A
COMDTOK  *         ENTRY  (X2) = 18/0,21/LEV OF IND ADDR,21/FWA          COMDTOK     358    A
COMDTOK  *                (X3) = 18/0,21/LEV OF IND ADDR,21/LWA          COMDTOK     359    A
COMDTOK  *                (X4) = 18/0,21/LEV OF IND ADDR,21/LEN          COMDTOK     360    A
COMDTOK  *                                                               COMDTOK     361    A
COMDTOK  *                (X3) IS OPTIONAL PARAMETER.  IF *LEN* IS .NZ., THEN  COMDTOK     362    A
COMDTOK  *                TOKEN DEASSEMBLY IS FROM FWA TO FWA+LEN-1.  ELSE  COMDTOK     363    A
COMDTOK  *                DEASSEMBLY IS FROM FWA TO LWA.                 COMDTOK     364    A
COMDTOK  *                                                               COMDTOK     365    A
COMDTOK  *         EXIT   (B2) = .MI. IF FWA, LWA, OR LEN IS BAD, ELSE .PL.  COMDTOK     366    A
COMDTOK  *                                                               COMDTOK     367    A
COMDTOK  *         USES   ALL                                            COMDTOK     368    A
COMDTOK  *                                                               COMDTOK     369    A
COMDTOK  *         CALLS  CAD,FLL,MCS,PRIDP,WOD,ZTB                      COMDTOK     370    A
COMDTOK                                                                  COMDTOK     371    A
COMDTOK                                                                  COMDTOK     372    A
COMDTOK   LTK      SUBR               ** ENTRY/EXIT **                   COMDTOK     373    A
COMDTOK            RJ     FLL         CHECK *FWA,LWA,LEN*                COMDTOK     374    A
COMDTOK            MI     B2,EXIT.    IF FWA, LWA, OR LEN IS BAD         COMDTOK     375    A
COMDTOK            BX6    X2                                             COMDTOK     376    A
COMDTOK            LX7    X3                                             COMDTOK     377    A
COMDTOK            SA6    LTK=FWA     SAVE FWA                           COMDTOK     378    A
COMDTOK            SA7    A6+1        SAVE LWA                           COMDTOK     379    A
COMDTOK                                                                  COMDTOK     380    A
COMDTOK  *         ** MAIN LOOP NODE **                                  COMDTOK     381    A
COMDTOK                                                                  COMDTOK     382    A
COMDTOK   LTK2     SA1    LTK=FWA     (X1) = ADDR OF NEXT WORD TO DUMP   COMDTOK     383    A
COMDTOK            SA2    A1+B1       (X2) = LWA TO DUMP                 COMDTOK     384    A
COMDTOK            IX3    X2-X1                                          COMDTOK     385    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMDTOK

```
COMDTOK          SX6    X1+B1                                          COMDTOK    386    A
COMDTOK          SB2    B0          SET TO *NO ERROR OCCURRED*         COMDTOK    387    A
COMDTOK          MI     X3,EXIT.    IF FINISHED                        COMDTOK    388    A
COMDTOK          SA6    A1                                             COMDTOK    389    A
COMDTOK                                                               COMDTOK    390    A
COMDTOK          RJ     CAD         CONVERT ADDR TO DPC                COMDTOK    391    A
COMDTOK          SA6    SNAPLNE                                        COMDTOK    392    A
COMDTOK          SA1    LTK=FWA                                        COMDTOK    393    A
COMDTOK          SA1    X1-1                                           COMDTOK    394    A
COMDTOK          RJ     =XWOD       CONVERT FULL WORD TO OCTAL DPC     COMDTOK    395    A
COMDTOK          SA6    A6+B1                                          COMDTOK    396    A
COMDTOK          SA7    A6+B1                                          COMDTOK    397    A
COMDTOK          SA1    LTK=FWA                                        COMDTOK    398    A
COMDTOK          SA1    X1-1                                           COMDTOK    399    A
COMDTOK          RJ     =XZTB       CONVERT ZEROES TO BLANKS           COMDTOK    400    A
COMDTOK          SA2    =5L                                            COMDTOK    401    A
COMDTOK          MX3    5*CHAR                                         COMDTOK    402    A
COMDTOK          BX7    X6                                             COMDTOK    403    A
COMDTOK          LX6    -5*CHAR     (X6) = FGHIJABCDE                  COMDTOK    404    A
COMDTOK          BX7    -X3*X7      (X7) = 00000FGHIJ                  COMDTOK    405    A
COMDTOK          BX6    -X3*X6      (X6) = 00000ABCDE                  COMDTOK    406    A
COMDTOK          IX7    X7+X2       (X7) = .....FGHIJ                  COMDTOK    407    A
COMDTOK          BX6    X6+X2       (X6) = .....ABCDE                  COMDTOK    408    A
COMDTOK          LX7    5*CHAR      (X7) = FGHIJ.....                  COMDTOK    409    A
COMDTOK          SA6    A7+B1                                          COMDTOK    410    A
COMDTOK          SA7    A6+B1                                          COMDTOK    411    A
COMDTOK          SB7    A7                                             COMDTOK    412    A
COMDTOK                                                               COMDTOK    413    A
COMDTOK          RJ     CTD         CONVERT TOKEN TYPE TO DPC          COMDTOK    414    A
COMDTOK          MI     B2,LTK3     IF USER DID NOT PROVIDE *CTD=*     COMDTOK    415    A
COMDTOK          RJ     =XMCS       MERGE CODED STRINGS                COMDTOK    416    A
COMDTOK                                                               COMDTOK    417    A
COMDTOK   LTK3   BX6    X6-X6                                          COMDTOK    418    A
COMDTOK          SA6    B7+B1       MARK EOL                           COMDTOK    419    A
COMDTOK          PRIDP  =XSNAPLNE                                      COMDTOK    420    A
COMDTOK          EQ     LTK2                                           COMDTOK    421    A
COMDTOK                                                               COMDTOK    422    A
COMDTOK                                                               COMDTOK    423    A
COMDTOK   LTK=FWA =     TOG=FWA                                       COMDTOK    424    A
COMDTOK   TID    SPACE  4,10                                           COMDTOK    425    A
COMDTOK  **      TID - *TOK* INTERACTIVE DEBUG INTERFACE.              COMDTOK    426    A
COMDTOK  *                                                             COMDTOK    427    A
COMDTOK  *                                                             COMDTOK    428    A
COMDTOK  *       I ORIGINALLY INTENDED THAT THIS ROUTINE WOULD BE MORE COMDTOK    429    A
COMDTOK  *       SOPHISTICATED THAN IT CURRENTLY IS...   MY INTENTION  COMDTOK    430    A
COMDTOK  *       WAS/IS --                                             COMDTOK    431    A
COMDTOK  *                                                             COMDTOK    432    A
COMDTOK  *         1. TO HAVE *TID* DISPLAY THE CURRENT *TOGEL* INSTRUCTION COMDTOK 433 A
COMDTOK  *            BEING EXECUTED.                                  COMDTOK    434    A
COMDTOK  *                                                             COMDTOK    435    A
COMDTOK  *         2. TO DISPLAY THE SOURCE LINE BEING ENTOKENED, INDICATING COMDTOK 436 A
COMDTOK  *            SOMEHOW WHERE WE CURRENTLY ARE IN THAT SOURCE LINE. COMDTOK  437    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMDTOK

```
COMDTOK  *                                                                  COMDTOK    438    A
COMDTOK  *          3. TO DISPLAY THE TOKEN(S) GENERATED VIA THIS TOGEL     COMDTOK    439    A
COMDTOK  *             INSTRUCTION.                                         COMDTOK    440    A
COMDTOK  *                                                                  COMDTOK    441    A
COMDTOK  *        HOWEVER, DUE TO TIME RESTRICTIONS RELATED (OF COURSE) TO  COMDTOK    442    A
COMDTOK  *        A LACK OF MANAGEMENT SUPPORT FOR TOOLS, WHAT YOU SEE IS   COMDTOK    443    A
COMDTOK  *        WHAT YOU GET.                                             COMDTOK    444    A
COMDTOK  *                                                                  COMDTOK    445    A
COMDTOK  *        ENTRY  ENTOKENING REGISTERS INTACT                        COMDTOK    446    A
COMDTOK  *                                                                  COMDTOK    447    A
COMDTOK  *        EXIT   ENTOKENING REGISTERS STILL INTACT                  COMDTOK    448    A
COMDTOK  *                                                                  COMDTOK    449    A
COMDTOK  *        USES   X - 1,7                                            COMDTOK    450    A
COMDTOK  *               A - 1                                              COMDTOK    451    A
COMDTOK  *               B - NONE                                           COMDTOK    452    A
COMDTOK  *                                                                  COMDTOK    453    A
COMDTOK  *        CALLS  BREAK(IDP)                                         COMDTOK    454    A
COMDTOK                                                                     COMDTOK    455    A
COMDTOK                                                                     COMDTOK    456    A
COMDTOK   TID     SUBR                ** ENTRY/EXIT **                      COMDTOK    457    A
COMDTOK                                                                     COMDTOK    458    A
COMDTOK  *        CHECK FOR A SPECIFICALLY SPECIFIED TOGEL INSTRUCTION.     COMDTOK    459    A
COMDTOK                                                                     COMDTOK    460    A
COMDTOK           SA1    =XTF=IDP    (X1) = SPECIFIED *TOGEL* INSTRUCTION   COMDTOK    461    A
COMDTOK           ZR     X1,EXIT.    IF NO BREAK                            COMDTOK    462    A
COMDTOK           MI     X1,TID2     IF UNCONDITIONAL BREAK                 COMDTOK    463    A
COMDTOK           BX7    X2                                                 COMDTOK    464    A
COMDTOK           LX7    0-TG.TOFUP                                         COMDTOK    465    A
COMDTOK           SX7    X7          (X7) = TOFU ADDR                       COMDTOK    466    A
COMDTOK           IX7    X1-X7                                              COMDTOK    467    A
COMDTOK           NZ     X7,EXIT.    IF NO MATCH, NO BREAK...               COMDTOK    468    A
COMDTOK                                                                     COMDTOK    469    A
COMDTOK   TID2    BSS    0                                                  COMDTOK    470    A
COMDTOK   BRTOK   BREAK                                                     COMDTOK    471    A
COMDTOK           EQ     EXIT.                                              COMDTOK    472    A
COMDTOK   TID     SPACE  4,10                                               COMDTOK    473    A
COMDTOK   TID=CAL =      *+4S15                                             COMDTOK    474    A
COMDTOK   TID=COF =      *+4S15                                             COMDTOK    475    A
COMDTOK   TID=ELS =      *+4S15                                             COMDTOK    476    A
COMDTOK   TID=ENT =      *+4S15                                             COMDTOK    477    A
COMDTOK   TID=GN  =      *+4S15                                             COMDTOK    478    A
COMDTOK   TID=GO  =      *+4S15                                             COMDTOK    479    A
COMDTOK   TID=GS  =      *+4S15                                             COMDTOK    480    A
COMDTOK   TID=IFT =      *+4S15                                             COMDTOK    481    A
COMDTOK   TID=ON  =      *+4S15                                             COMDTOK    482    A
COMDTOK   TID=XEQ =      *+4S15                                             COMDTOK    483    A
COMDTOK   TOD     SPACE  4,10                                               COMDTOK    484    A
COMDTOK  **       TOD - *TOGEL* DEASSEMBLER.                                COMDTOK    485    A
COMDTOK  *                                                                  COMDTOK    486    A
COMDTOK  *                                                                  COMDTOK    487    A
COMDTOK  *        THIS ROUTINE DEASSEMBLES A SINGLE *TOGEL* BINARY INSTRUCTION  COMDTOK  488  A
COMDTOK  *        AND PRINTS IT VIA *PRIDP*  (*COMCIDP* INTERFACE).         COMDTOK    489    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMDTOK

```
COMDTOK  *                                                                    COMDTOK    490    A
COMDTOK  *         ENTRY   (X1) = ADDR OF *TOGEL* INSTRUCTION TO DEASSEMBLE   COMDTOK    491    A
COMDTOK  *                                                                    COMDTOK    492    A
COMDTOK  *         EXIT    NONE                                               COMDTOK    493    A
COMDTOK  *                                                                    COMDTOK    494    A
COMDTOK  *         USES    ALL BUT A0,X0                                      COMDTOK    495    A
COMDTOK  *                                                                    COMDTOK    496    A
COMDTOK  *         CALLS   IN *COMDTOK* -- DCR                                COMDTOK    497    A
COMDTOK  *                 IN *COMCIDP* -- CAD,CHK,PRIDP                      COMDTOK    498    A
COMDTOK  *                 OTHERS       -- MCS,WOD,ZTB                        COMDTOK    499    A
COMDTOK                                                                       COMDTOK    500    A
COMDTOK                                                                       COMDTOK    501    A
COMDTOK   TOD      SUBR                ** ENTRY/EXIT **                       COMDTOK    502    A
COMDTOK                                                                       COMDTOK    503    A
COMDTOK  *         SET UP ADDR OF *TOGEL* BINARY INSTRUCTION TO DEASSEMBLE.   COMDTOK    504    A
COMDTOK                                                                       COMDTOK    505    A
COMDTOK            SX5     X1          SAVE (X1) = ADDR OF *TOGEL* INST       COMDTOK    506    A
COMDTOK            RJ      CAD         CONVERT ADDRESS                        COMDTOK    507    A
COMDTOK            SA6     =XSNAPLNE                                          COMDTOK    508    A
COMDTOK                                                                       COMDTOK    509    A
COMDTOK  *         CONVERT BINARY *TOGEL* INSTRUCTION TO OCTAL DPC.           COMDTOK    510    A
COMDTOK                                                                       COMDTOK    511    A
COMDTOK            SA1     X5          (A1,X1) = A+C OF *TOGEL* INST TO DEASSEMBLE COMDTOK 512  A
COMDTOK            RJ      =XWOD       CONVERT A FULL BINARY WORD TO OCTAL DPC COMDTOK    513    A
COMDTOK            SA6     A6+B1                                              COMDTOK    514    A
COMDTOK            SA7     A6+B1                                              COMDTOK    515    A
COMDTOK                                                                       COMDTOK    516    A
COMDTOK            SA2     =10H                                               COMDTOK    517    A
COMDTOK            BX6     X2                                                 COMDTOK    518    A
COMDTOK            SA6     A7+B1                                              COMDTOK    519    A
COMDTOK                                                                       COMDTOK    520    A
COMDTOK  *         SEARCH FOR *TOGEL* BINARY OPCODE (I.E. *COMCTOK/TOFU* ADDR) COMDTOK   521    A
COMDTOK  *         IN OPCODE TABLE.                                           COMDTOK    522    A
COMDTOK                                                                       COMDTOK    523    A
COMDTOK            SA5     A1          (A5,X5) = A+C OF *TOGEL* INST TO DEASSEMBLE COMDTOK 524  A
COMDTOK            LX5     0-TG.TOFUP                                         COMDTOK    525    A
COMDTOK            SB7     L.OPC       (B7) = LEN OF *TOGEL* OPCODE TABLE     COMDTOK    526    A
COMDTOK            SA1     FW.OPC      (A1,X1) = A+C OF 1ST ENTRY IN OPCODE TABLE COMDTOK 527  A
COMDTOK            SX6     X5          (X6) = ADDR TO SEARCH FOR              COMDTOK    528    A
COMDTOK                                                                       COMDTOK    529    A
COMDTOK   TOD2     LE      B7,B0,TOD=DAT IF END-OF-TABLE AND NO FIND, DATA... COMDTOK    530    A
COMDTOK            BX2     X1                                                 COMDTOK    531    A
COMDTOK            SX1     X1                                                 COMDTOK    532    A
COMDTOK            BX7     X6-X1                                              COMDTOK    533    A
COMDTOK            SA1     A1+2                                               COMDTOK    534    A
COMDTOK            SB7     B7-2                                               COMDTOK    535    A
COMDTOK            NZ      X7,TOD2     IF NO FIND, CONTINUE...                COMDTOK    536    A
COMDTOK                                                                       COMDTOK    537    A
COMDTOK  *         HERE IF A FIND.                                            COMDTOK    538    A
COMDTOK  *                                                                    COMDTOK    539    A
COMDTOK  *           1. EXTRACT APPROPRIATE *TOD=* DEASSEMBLER ADDR           COMDTOK    540    A
COMDTOK  *              FOR THIS INSTRUCTION.                                 COMDTOK    541    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMDTOK

```
COMDTOK  *                                                                    COMDTOK    542    A
COMDTOK  *          2. EXTRACT APPROPRIATE OPCODE FOR THIS INSTRUCTION.       COMDTOK    543    A
COMDTOK                                                                       COMDTOK    544    A
COMDTOK            LX2    -18                                                 COMDTOK    545    A
COMDTOK            SB5    X2          (B5) = *TOD=* ADDR FOR THIS INSTRUCTION  COMDTOK    546    A
COMDTOK            SA1    A1-B1       (X1) = OPCODE FOR THIS INST, -L- FMT     COMDTOK    547    A
COMDTOK            BX6    X1                                                  COMDTOK    548    A
COMDTOK            SA6    A6+B1                                               COMDTOK    549    A
COMDTOK            SB7    A6                                                  COMDTOK    550    A
COMDTOK            SX1    1R                                                  COMDTOK    551    A
COMDTOK            RJ     =XMCS       MERGE CODED STRINGS                     COMDTOK    552    A
COMDTOK            JP     B5+                                                 COMDTOK    553    A
COMDTOK   TOD=CAL  SPACE  4,10                                                COMDTOK    554    A
COMDTOK  **        HERE IF *CALT XX*.                                         COMDTOK    555    A
COMDTOK                                                                       COMDTOK    556    A
COMDTOK   TOD=CAL  BSS    0                                                   COMDTOK    557    A
COMDTOK            LX5    0-TG.CADP+TG.TOFUP-0                                COMDTOK    558    A
COMDTOK            SX1    X5          (X1) = ADDR OF *CALT* PROCESSOR         COMDTOK    559    A
COMDTOK            RJ     =XCAD       CONVERT ADDRESS                         COMDTOK    560    A
COMDTOK            SB7    A6          RESTORE (B7)                            COMDTOK    561    A
COMDTOK            BX1    X6                                                  COMDTOK    562    A
COMDTOK            RJ     =XMCS       MERGE CODED STRINGS                     COMDTOK    563    A
COMDTOK            EQ     TOD=X                                               COMDTOK    564    A
COMDTOK   TOD=COF  SPACE  4,10                                                COMDTOK    565    A
COMDTOK  **        HERE IF *CASEOF (A..Z)*.                                   COMDTOK    566    A
COMDTOK                                                                       COMDTOK    567    A
COMDTOK   TOD=COF  BSS    0                                                   COMDTOK    568    A
COMDTOK            SX1    1R(                                                 COMDTOK    569    A
COMDTOK            RJ     =XMCS       MERGE CODED STRINGS                     COMDTOK    570    A
COMDTOK            LX5    0-TG.COAP+TG.TOFUP-0                                COMDTOK    571    A
COMDTOK            MX2    -TG.COAL                                            COMDTOK    572    A
COMDTOK            BX1    -X2*X5      (X1) = 1ST CHAR IN *CASEOF* RANGE,      COMDTOK    573    A
COMDTOK  *                                   DPC IN -R- FMT                   COMDTOK    574    A
COMDTOK            RJ     =XMCS       MERGE CODED STRINGS                     COMDTOK    575    A
COMDTOK            SX1    2R..                                                COMDTOK    576    A
COMDTOK            RJ     =XMCS       MERGE CODED STRINGS                     COMDTOK    577    A
COMDTOK            LX5    0-TG.COZP+TG.COAP-0                                 COMDTOK    578    A
COMDTOK            MX2    -TG.COZL                                            COMDTOK    579    A
COMDTOK            BX3    -X2*X5      (X3) = LAST CHAR IN *CASEOF* RANGE,     COMDTOK    580    A
COMDTOK  *                                   IN *RCS* REPRESENTATION, -R- FMT COMDTOK    581    A
COMDTOK            SX4    X1+10000B+BICH                                      COMDTOK    582    A
COMDTOK            BX1    -X2*X4      (X1) = LAST CHAR IN *CASEOF* RANGE,     COMDTOK    583    A
COMDTOK  *                                   IN DPC, -R- FMT                  COMDTOK    584    A
COMDTOK            RJ     =XMCS       MERGE CODED STRINGS                     COMDTOK    585    A
COMDTOK            SX1    1R)                                                 COMDTOK    586    A
COMDTOK            RJ     =XMCS       MERGE CODED STRINGS                     COMDTOK    587    A
COMDTOK            EQ     TOD=X                                               COMDTOK    588    A
COMDTOK   TOD=DAT  SPACE  4,10                                                COMDTOK    589    A
COMDTOK  **        HERE IF *DATA*.                                            COMDTOK    590    A
COMDTOK                                                                       COMDTOK    591    A
COMDTOK   TOD=DAT  BSS    0                                                   COMDTOK    592    A
COMDTOK            SA1    =0LDATA                                             COMDTOK    593    A
```

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMDTOK

```
COMDTOK            BX6    X1                                                  COMDTOK     594    A
COMDTOK            SA6    A6+B1                                               COMDTOK     595    A
COMDTOK            SB7    A6                                                  COMDTOK     596    A
COMDTOK            EQ     TOD=X                                               COMDTOK     597    A
COMDTOK   TOD=ELS  SPACE  4,10                                                COMDTOK     598    A
COMDTOK  **        HERE IF --                                                 COMDTOK     599    A
COMDTOK  *                                                                    COMDTOK     600    A
COMDTOK  *           1. ELST                                                  COMDTOK     601    A
COMDTOK  *           2. ENDT                                                  COMDTOK     602    A
COMDTOK                                                                       COMDTOK     603    A
COMDTOK   TOD=ELS  BSS    0                                                   COMDTOK     604    A
COMDTOK   TOD=ENT  BSS    0                                                   COMDTOK     605    A
COMDTOK            LX5    TG.TOFUP-0                                          COMDTOK     606    A
COMDTOK            MX1    7*CHAR                                              COMDTOK     607    A
COMDTOK            BX1    X1*X5       (X1) = LABEL FOR THIS CONDITIONAL OPCODE COMDTOK     608    A
COMDTOK            RJ     =XZTB       CONVERT ZEROES TO BLANKS                COMDTOK     609    A
COMDTOK            LX6    -CHAR                                               COMDTOK     610    A
COMDTOK            SA6    A6-B1                                               COMDTOK     611    A
COMDTOK            EQ     TOD=X                                               COMDTOK     612    A
COMDTOK   TOD=GN   SPACE  4,10                                                COMDTOK     613    A
COMDTOK  **        HERE IF *GROUP (..),TOT,NSQZ*.                             COMDTOK     614    A
COMDTOK                                                                       COMDTOK     615    A
COMDTOK   TOD=GN   BSS    0                                                   COMDTOK     616    A
COMDTOK            LX5    0-TG.MXAP+TG.TOFUP-0                                COMDTOK     617    A
COMDTOK            SA4    X5          (X4) = CHAR SHIFT MASK                  COMDTOK     618    A
COMDTOK            RJ     DCR         DASSEMBLE CHAR RANGE SHIFT MASK         COMDTOK     619    A
COMDTOK            SX1    1R,                                                 COMDTOK     620    A
COMDTOK            LX1    10*CHAR-1*CHAR                                      COMDTOK     621    A
COMDTOK            RJ     =XMCS       MERGE CODED STRINGS                     COMDTOK     622    A
COMDTOK            LX5    0-TG.TOTP+TG.MXAP-0                                 COMDTOK     623    A
COMDTOK                                                                       COMDTOK     624    A
COMDTOK            SX1    X5          (X1) = TOKEN TYPE                       COMDTOK     625    A
COMDTOK            RJ     CTD         CONVERT TOKEN TYPE TO DPC               COMDTOK     626    A
COMDTOK            RJ     =XMCS       MERGE CODED STRINGS                     COMDTOK     627    A
COMDTOK                                                                       COMDTOK     628    A
COMDTOK            SA1    =L.,NSQZ.                                           COMDTOK     629    A
COMDTOK            RJ     =XMCS       MERGE CODED STRINGS                     COMDTOK     630    A
COMDTOK            EQ     TOD=X                                               COMDTOK     631    A
COMDTOK   TOD=GO   SPACE  4,10                                                COMDTOK     632    A
COMDTOK  **        HERE IF *GOTO XX*.                                         COMDTOK     633    A
COMDTOK                                                                       COMDTOK     634    A
COMDTOK   TOD=GO   BSS    0                                                   COMDTOK     635    A
COMDTOK            LX5    0-TG.GOTP+TG.TOFUP-0                                COMDTOK     636    A
COMDTOK            SX1    X5          (X1) = ADDR TO *GOTO* TO                COMDTOK     637    A
COMDTOK            RJ     =XCAD       CONVERT ADDRESS                         COMDTOK     638    A
COMDTOK            SB7    A6          RESTORE (B7)                            COMDTOK     639    A
COMDTOK            BX1    X6                                                  COMDTOK     640    A
COMDTOK            RJ     =XMCS       MERGE CODED STRINGS                     COMDTOK     641    A
COMDTOK            EQ     TOD=X                                               COMDTOK     642    A
COMDTOK   TOD=GS   SPACE  4,10                                                COMDTOK     643    A
COMDTOK  **        HERE IF *GROUP (..),TOT,SQZ*.                              COMDTOK     644    A
COMDTOK                                                                       COMDTOK     645    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMDTOK

```
COMDTOK   TOD=GS   BSS    0                                                COMDTOK    646    A
COMDTOK            LX5    0-TG.MXAP+TG.TOFUP-0                             COMDTOK    647    A
COMDTOK            SA4    X5          (X4) = CHAR SHIFT MASK               COMDTOK    648    A
COMDTOK            RJ     DCR         DASSEMBLE CHAR RANGE SHIFT MASK      COMDTOK    649    A
COMDTOK            SX1    1R,                                              COMDTOK    650    A
COMDTOK            LX1    10*CHAR-1*CHAR                                   COMDTOK    651    A
COMDTOK            RJ     =XMCS       MERGE CODED STRINGS                  COMDTOK    652    A
COMDTOK            LX5    0-TG.TOTP+TG.MXAP-0                              COMDTOK    653    A
COMDTOK                                                                    COMDTOK    654    A
COMDTOK            SX1    X5          (X1) = TOKEN TYPE                    COMDTOK    655    A
COMDTOK            RJ     CTD         CONVERT TOKEN TYPE TO DPC            COMDTOK    656    A
COMDTOK            RJ     =XMCS       MERGE CODED STRINGS                  COMDTOK    657    A
COMDTOK                                                                    COMDTOK    658    A
COMDTOK            SA1    =L.,SQZ.                                         COMDTOK    659    A
COMDTOK            RJ     =XMCS       MERGE CODED STRINGS                  COMDTOK    660    A
COMDTOK            EQ     TOD=X                                            COMDTOK    661    A
COMDTOK   TOD=IFT  SPACE  4,10                                             COMDTOK    662    A
COMDTOK   **       HERE IF *IFT (..)*.                                     COMDTOK    663    A
COMDTOK                                                                    COMDTOK    664    A
COMDTOK   TOD=IFT  BSS    0                                                COMDTOK    665    A
COMDTOK            LX5    0-TG.MXAP+TG.TOFUP-0                             COMDTOK    666    A
COMDTOK            SA4    X5          (X4) = CHAR SHIFT MASK               COMDTOK    667    A
COMDTOK            RJ     DCR         DEASSEMBLE CHAR RANGE SHIFT MASK     COMDTOK    668    A
COMDTOK            EQ     TOD=X                                            COMDTOK    669    A
COMDTOK   TOD=ON   SPACE  4,10                                             COMDTOK    670    A
COMDTOK   **       HERE IF *ON I,V*.                                       COMDTOK    671    A
COMDTOK                                                                    COMDTOK    672    A
COMDTOK                                                                    COMDTOK    673    A
COMDTOK   TOD=ON   BSS    0                                                COMDTOK    674    A
COMDTOK            LX5    0-TG.INCP+TG.TOFUP-0                             COMDTOK    675    A
COMDTOK            SX1    X5          (X1) = ADDR OF INTERUPT CELL         COMDTOK    676    A
COMDTOK            RJ     =XCAD       CONVERT ADDRESS                      COMDTOK    677    A
COMDTOK            SB7    A6          RESTORE (B7)                         COMDTOK    678    A
COMDTOK            LX6    10*CHAR-6*CHAR                                   COMDTOK    679    A
COMDTOK            MX2    6*CHAR                                           COMDTOK    680    A
COMDTOK            BX1    X2*X6       (X1) = ADDR OF INTERUPT CELL, -L- FMT COMDTOK   681    A
COMDTOK            RJ     =XMCS       MERGE CODED STRINGS                  COMDTOK    682    A
COMDTOK            SX1    1R,                                              COMDTOK    683    A
COMDTOK            RJ     =XMCS       MERGE CODED STRINGS                  COMDTOK    684    A
COMDTOK            SA2    B7                                               COMDTOK    685    A
COMDTOK            LX5    0-TG.INVP+TG.INCP-0                              COMDTOK    686    A
COMDTOK            BX6    X2                                               COMDTOK    687    A
COMDTOK            SX1    X5          (X1) = VALUE FOR INTERUPT CELL       COMDTOK    688    A
COMDTOK            SA6    B7          SAVE (B7) IN (A6)                    COMDTOK    689    A
COMDTOK            RJ     =XCAD       CONVERT ADDRESS (I.E. **PROBABLY** AN ADDR) COMDTOK 690  A
COMDTOK            SB7    A6          RESTORE (B7)                         COMDTOK    691    A
COMDTOK            LX6    10*CHAR-6*CHAR                                   COMDTOK    692    A
COMDTOK            MX2    6*CHAR                                           COMDTOK    693    A
COMDTOK            BX1    X2*X6       (X1) = INTERUPT VALUE, -L- FORMAT    COMDTOK    694    A
COMDTOK            RJ     =XMCS       MERGE CODED STRINGS                  COMDTOK    695    A
COMDTOK            EQ     TOD=X                                            COMDTOK    696    A
COMDTOK                                                                    COMDTOK    697    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMDTOK

```
COMDTOK   TOD=XEQ  SPACE  4,10                                          COMDTOK     698    A
COMDTOK  **        HERE IF *XEQ P*.                                     COMDTOK     699    A
COMDTOK                                                                 COMDTOK     700    A
COMDTOK                                                                 COMDTOK     701    A
COMDTOK   TOD=XEQ  BSS    0                                             COMDTOK     702    A
COMDTOK            EQ     TOD=X       (FOR NOW, DO NOTHING...)          COMDTOK     703    A
COMDTOK   TOD=X    SPACE  4,10                                          COMDTOK     704    A
COMDTOK  **        HERE WHEN FINISHED DEASSEMBLING *TOGEL* INSTRUCTION, COMDTOK     705    A
COMDTOK  *         TIME TO PRINT LINE VIA *COMCIDP*.                    COMDTOK     706    A
COMDTOK                                                                 COMDTOK     707    A
COMDTOK   TOD=X    BSS    0                                             COMDTOK     708    A
COMDTOK            BX6    X6-X6                                         COMDTOK     709    A
COMDTOK            SA6    B7+B1       MARK EOL                          COMDTOK     710    A
COMDTOK            PRIDP  =XSNAPLNE                                     COMDTOK     711    A
COMDTOK            SB2    B0          SET TO *NO ERROR*                 COMDTOK     712    A
COMDTOK            EQ     EXIT.                                         COMDTOK     713    A
COMDTOK   UTG      SPACE  4,10                                          COMDTOK     714    A
COMDTOK  **        UTG - USER *TOGEL* OWNCODE.                          COMDTOK     715    A
COMDTOK  *                                                              COMDTOK     716    A
COMDTOK  *                                                              COMDTOK     717    A
COMDTOK  *         ENTRY  NONE                                          COMDTOK     718    A
COMDTOK  *                                                              COMDTOK     719    A
COMDTOK  *         EXIT   NONE                                          COMDTOK     720    A
COMDTOK  *                                                              COMDTOK     721    A
COMDTOK  *         USES   CANNOT USE A5,X5,A0,X0                        COMDTOK     722    A
COMDTOK  *                                                              COMDTOK     723    A
COMDTOK  *         CALLS  UTG=(IF AVAIL)                                COMDTOK     724    A
COMDTOK                                                                 COMDTOK     725    A
COMDTOK                                                                 COMDTOK     726    A
COMDTOK   UTG      SUBR              ** ENTRY/EXIT **                   COMDTOK     727    A
COMDTOK            SB2    =YUTG=                                        COMDTOK     728    A
COMDTOK            MI     B2,EXIT.    IF USER DID NOT PROVIDE *UTG=*    COMDTOK     729    A
COMDTOK            RJ     =YUTG=      USER *TOGEL* OWNCODE              COMDTOK     730    A
COMDTOK            EQ     EXIT.                                         COMDTOK     731    A
COMDTOK   UTK      SPACE  4,10                                          COMDTOK     732    A
COMDTOK  **        UTK - USER *TOKEN* OWNCODE.                          COMDTOK     733    A
COMDTOK  *                                                              COMDTOK     734    A
COMDTOK  *                                                              COMDTOK     735    A
COMDTOK  *         ENTRY  NONE                                          COMDTOK     736    A
COMDTOK  *                                                              COMDTOK     737    A
COMDTOK  *         EXIT   NONE                                          COMDTOK     738    A
COMDTOK  *                                                              COMDTOK     739    A
COMDTOK  *         USES   CANNOT USE A5,X5,A0,X0                        COMDTOK     740    A
COMDTOK  *                                                              COMDTOK     741    A
COMDTOK  *         CALLS  UTK=(IF AVAIL)                                COMDTOK     742    A
COMDTOK                                                                 COMDTOK     743    A
COMDTOK                                                                 COMDTOK     744    A
COMDTOK   UTK      SUBR              ** ENTRY/EXIT **                   COMDTOK     745    A
COMDTOK            SB2    =YUTK=                                        COMDTOK     746    A
COMDTOK            MI     B2,EXIT.    IF USER DID NOT PROVIDE *UTK=*    COMDTOK     747    A
COMDTOK            RJ     =YUTK=      USER *TOKEN* OWNCODE              COMDTOK     748    A
COMDTOK            EQ     EXIT.                                         COMDTOK     749    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMDTOK

COMDTOK   TOK      SPACE  4,10                                          COMDTOK      750     A
COMDTOK   TOK=IDP  =      TID                                           COMDTOK      751     A
COMDTOK                                                                 COMDTOK      752     A
COMDTOK                                                                 COMDTOK      753     A
COMDTOK   QUAL$    IF     -DEF,QUAL$                                    COMDTOK      754     A
COMDTOK            QUAL   *                                             COMDTOK      755     A
COMDTOK   LTK      =      /COMDTOK/LTK                                  COMDTOK      756     A
COMDTOK   TOK=IDP  =      /COMDTOK/TOK=IDP                              COMDTOK      757     A
COMDTOK   TOK      =      /COMDTOK/TOD                                  COMDTOK      758     A
COMDTOK   ST=TOG   =      /COMDTOK/ST=TOG                               COMDTOK      759     A
COMDTOK   ST=TOK   =      /COMDTOK/ST=TOK                               COMDTOK      760     A
COMDTOK   QUAL$    ENDIF                                                COMDTOK      761     A
COMDTOK   COMDTOK  ENDX                                                 COMDTOK      762     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - COMDTOK

          IDENTIFIER                TOTAL    ACTIVE

          COMDTOK                     762       762

          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMQSVR

COMQSVR  *COMDECK COMQSVR                                               COMQSVR        1     A
COMQSVR   SVR=     CTEXT  COMQSVR - SAVE AND RESTORE REGISTERS.         COMQSVR        2     A
COMQSVR   SVR=     SPACE  4,8                                           COMQSVR        3     A
COMQSVR            IF     -DEF,QUAL$,1                                  COMQSVR        4     A
COMQSVR            QUAL   COMQSVR                                       COMQSVR        5     A
COMQSVR            BASE   D                                             COMQSVR        6     A
COMQSVR   SVR=     SPACE  4,8                                           COMQSVR        7     A
COMQSVR  ***       SVR - SAVE AND RESTORE REGISTERS.                    COMQSVR        8     A
COMQSVR  *                                                              COMQSVR        9     A
COMQSVR  *                                                              COMQSVR       10     A
COMQSVR  *                CONTAINS THE ROUTINES *SVR=* TO SAVE ALL REGISTERS,  COMQSVR  11     A
COMQSVR  *         AND *RSR=* TO RESTORE ALL REGISTERS. MINUS ZERO (-0) IS PRE-  COMQSVR  12     A
COMQSVR  *         SERVED IN ALL REGISTERS, PARTICULARLY -A- AND -B- REGISTERS.  COMQSVR  13     A
COMQSVR  *         EACH REGISTER IS SAVED IN A FULL CM WORD (-A- AND -B-         COMQSVR  14     A
COMQSVR  *         REGISTERS ARE SIGN EXTENDED TO BIT 59). A REGISTER BLOCK IS   COMQSVR  15     A
COMQSVR  *         DEFINED TO BE ALL THE -B- REGISTERS, ALL THE -A- REGISTERS, OR COMQSVR  16     A
COMQSVR  *         ALL THE -X- REGISTERS. ALL REGISTERS IN A REGISTER BLOCK ARE  COMQSVR  17     A
COMQSVR  *         SAVED CONTIGUOUSLY IN ASCENDING ORDER.                       COMQSVR  18     A
COMQSVR  *                                                              COMQSVR       19     A
COMQSVR  *         ASSEMBLY-TIME OPTIONS--                              COMQSVR       20     A
COMQSVR  *                                                              COMQSVR       21     A
COMQSVR  *            IF THE SYMBOL *BN=1* (N=1 THRU 7) IS DEFINED, (B.N) ARE  COMQSVR  22     A
COMQSVR  *         ASSUMED TO BE EQUAL TO 1. IF *BN=1* IS NOT DEFINED, (B1)     COMQSVR  23     A
COMQSVR  *         ARE CHECKED FOR 1 TO TRY TO AVOID EXECUTING THE *SAVE (B1)   COMQSVR  24     A
COMQSVR  *         VIA RJ SEQUENCE* CODE.                               COMQSVR       25     A
COMQSVR  *                                                              COMQSVR       26     A
COMQSVR  *            IF THE SYMBOL *#SVR* IS DEFINED (DEF), THEN THE CALLER  COMQSVR  27     A
COMQSVR  *         WILL PROVIDE THE FWA OF THE REGISTER SAVE AREA IN THE LOWER  COMQSVR  28     A
COMQSVR  *         30 BITS OF THE CALLING *RJ* FOR *SVR=*--             COMQSVR       29     A
COMQSVR  *                30/RJ SVR, 30/FWA OF REGISTER SAVE AREA       COMQSVR       30     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMQSVR

```
COMQSVR  *         AND IN (X1) FOR *RSR=*.                                          COMQSVR      31    A
COMQSVR  *            IF THE SYMBOL *#SVR* IS NOT DEFINED (-DEF), THEN A LOCAL      COMQSVR      32    A
COMQSVR  *         REGISTER SAVE AREA, (SVREG) ET SEQ, WILL BE USED AND NO          COMQSVR      33    A
COMQSVR  *         PARAMETERS NEED TO BE PASSED.                                    COMQSVR      34    A
COMQSVR  *                                                                          COMQSVR      35    A
COMQSVR  *            THE MICRO "SVBAX" DEFINES THE ORDER IN WHICH THE -B-, -A-,    COMQSVR      36    A
COMQSVR  *         AND -X- REGISTER BLOCKS ARE SAVED IN CM.                         COMQSVR      37    A
COMQSVR  *         I.E.   SVBAX    MICRO  1,,/BAX/        (DEFAULT) MEANS           COMQSVR      38    A
COMQSVR  *                SV=B     BSSZ   8                                         COMQSVR      39    A
COMQSVR  *                SV=A     BSSZ   8                                         COMQSVR      40    A
COMQSVR  *                SV=X     BSSZ   8                                         COMQSVR      41    A
COMQSVR  *         OR     SVBAX    MICRO  1,,/ABX/        MEANS                     COMQSVR      42    A
COMQSVR  *                SV=A     BSSZ   8                                         COMQSVR      43    A
COMQSVR  *                SV=B     BSSZ   8                                         COMQSVR      44    A
COMQSVR  *                SV=X     BSSZ   8                                         COMQSVR      45    A
COMQSVR   SVR=     SPACE  4,8                                                       COMQSVR      46    A
COMQSVR            IF     -MIC,SVBAX,1                                              COMQSVR      47    A
COMQSVR   SVBAX    MICRO  1,,/BAX/                                                  COMQSVR      48    A
COMQSVR                                                                             COMQSVR      49    A
COMQSVR   .1       MICRO  1,1,/"SVBAX"/                                             COMQSVR      50    A
COMQSVR   .2       MICRO  2,1,/"SVBAX"/                                             COMQSVR      51    A
COMQSVR   .3       MICRO  3,1,/"SVBAX"/                                             COMQSVR      52    A
COMQSVR                                                                             COMQSVR      53    A
COMQSVR   SV".1"   =      0                                                         COMQSVR      54    A
COMQSVR   SV".2"   =      0+8                                                       COMQSVR      55    A
COMQSVR   SV".3"   =      0+8+8                                                     COMQSVR      56    A
COMQSVR                                                                             COMQSVR      57    A
COMQSVR   .SVR     IF     -DEF,#SVR                                                COMQSVR      58    A
COMQSVR   SVREG    BSSZ   3*8                                                      COMQSVR      59    A
COMQSVR                                                                             COMQSVR      60    A
COMQSVR   .SVR     ELSE                                                             COMQSVR      61    A
COMQSVR   SVX7     BSSZ   1           TEMPORARY SAVE (X7)                           COMQSVR      62    A
COMQSVR   SVA5     BSSZ   1           TEMPORARY SAVE (A5)                           COMQSVR      63    A
COMQSVR   SVX5     BSSZ   1           TEMPORARY SAVE (X5)                           COMQSVR      64    A
COMQSVR   .SVR     ENDIF                                                            COMQSVR      65    A
COMQSVR                                                                             COMQSVR      66    A
COMQSVR   SVR=     SPACE  4,8                                                       COMQSVR      67    A
COMQSVR  **        SVR - SAVE ALL REGISTERS.                                        COMQSVR      68    A
COMQSVR  *                                                                          COMQSVR      69    A
COMQSVR  *                                                                          COMQSVR      70    A
COMQSVR  *         ENTRY  IF *#SVR* DEFINED--                                       COMQSVR      71    A
COMQSVR  *                LOWER 30 BITS OF CALLING *RJ* CONTAINS FWA OF             COMQSVR      72    A
COMQSVR  *                 REGISTER SAVE AREA (RSA)--   30/RJ SVR=, 30/FWA OF RSA  COMQSVR      73    A
COMQSVR  *                                                                          COMQSVR      74    A
COMQSVR  *                IF *#SVR* NOT DEFINED-- NONE                              COMQSVR      75    A
COMQSVR  *                                                                          COMQSVR      76    A
COMQSVR  *                IF *BN=1* DEFINED-- (B.N) = 1                             COMQSVR      77    A
COMQSVR  *                                                                          COMQSVR      78    A
COMQSVR  *         EXIT   (B1) = 1                                                  COMQSVR      79    A
COMQSVR  *                                                                          COMQSVR      80    A
COMQSVR  *         USES   ALL                                                       COMQSVR      81    A
COMQSVR  *                                                                          COMQSVR      82    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMQSVR

```
COMQSVR  *         CALLS  NONE                                                  COMQSVR     83   A
COMQSVR                                                                         COMQSVR     84   A
COMQSVR                                                                         COMQSVR     85   A
COMQSVR   SVR=     EQ     *+4S15      ** ENTRY/EXIT **                          COMQSVR     86   A
COMQSVR                                                                         COMQSVR     87   A
COMQSVR  *         CHECK TO SEE IF ANY REGISTER HAS BEEN DEFINED TO BE EQUAL    COMQSVR     88   A
COMQSVR  *           TO 1 AT ASSEMBLY TIME VIA *BN=1* SYMBOL. IF NOT, CHECK FOR  COMQSVR     89   A
COMQSVR  *           (B1) = 1. IF (B1).NE.1, SAVE (B1) THE HARD WAY BY EXECUTING COMQSVR     90   A
COMQSVR  *           THE *SAVE (B1)* RJ SEQUENCE AT *SVR2*. IF BIT 2**N WAS ON   COMQSVR     91   A
COMQSVR  *           IN (B1), THEN THE *RJ* AT (SVR2+2*N+1) WILL BE EXECUTED,    COMQSVR     92   A
COMQSVR  *           RESULTING IN AN /EQ SVR2+2*N+2/. IF BIT 2**N WAS NOT ON     COMQSVR     93   A
COMQSVR  *           IN (B1), THEN THE *RJ* WILL NOT BE  EXECUTED. BY QUERYING   COMQSVR     94   A
COMQSVR  *           (SVR2+2*N+1), ONE CAN DETERMINE WHETHER OR NOT BIT 2**N     COMQSVR     95   A
COMQSVR  *           WAS ON IN (B1) BY THE PRESENCE OF AN *EQ* (BIT WAS ON), OR  COMQSVR     96   A
COMQSVR  *           AN *RJ* (BIT WAS OFF) INSTRUCTION.                          COMQSVR     97   A
COMQSVR                                                                         COMQSVR     98   A
COMQSVR   B=1      MICRO  1,,/B0/                                               COMQSVR     99   A
COMQSVR            ECHO   4,BN=(B1,B7,B5,B2,B3,B4,B6)                           COMQSVR    100   A
COMQSVR            IF     DEF,BN=1,3                                            COMQSVR    101   A
COMQSVR   B=1      MICRO  1,,/BN/                                               COMQSVR    102   A
COMQSVR            STOPDUP                                                      COMQSVR    103   A
COMQSVR   B.EQ.1   SKIP                                                         COMQSVR    104   A
COMQSVR                                                                         COMQSVR    105   A
COMQSVR            LE     B1,B0,SVR2  IF (B1).LE.0 (INSURES -0 WILL BE HONORED) COMQSVR    106   A
COMQSVR            SB1    B1-1                                                  COMQSVR    107   A
COMQSVR            ZR     B1,SVR4     IF (B1) = 1                               COMQSVR    108   A
COMQSVR            SB1    B1+1        RESTORE (B1)                              COMQSVR    109   A
COMQSVR                                                                         COMQSVR    110   A
COMQSVR   SVR2     BSS    0                                                     COMQSVR    111   A
COMQSVR            PL     B1,*+2                                                COMQSVR    112   A
COMQSVR   +        RJ     *                                                     COMQSVR    113   A
COMQSVR                                                                         COMQSVR    114   A
COMQSVR   B.NE.1   DUP    17                                                    COMQSVR    115   A
COMQSVR   +        SB1    B1+B1                                                 COMQSVR    116   A
COMQSVR            NO                                                           COMQSVR    117   A
COMQSVR            PL     B1,*+2                                                COMQSVR    118   A
COMQSVR   +        RJ     *                                                     COMQSVR    119   A
COMQSVR   B.NE.1   ENDD                                                         COMQSVR    120   A
COMQSVR                                                                         COMQSVR    121   A
COMQSVR   +        SB0    B0+         FILLER BECAUSE /RJ */S MUST BE 2 WORDS    COMQSVR    122   A
COMQSVR  *                               APART FOR RESTORE                      COMQSVR    123   A
COMQSVR                                                                         COMQSVR    124   A
COMQSVR   SVR3     RJ     *           (B1)=1/(B1).NE.1 FLAG                     COMQSVR    125   A
COMQSVR                                                                         COMQSVR    126   A
COMQSVR   SVR4     BSS    0                                                     COMQSVR    127   A
COMQSVR                                                                         COMQSVR    128   A
COMQSVR   B.EQ.1   ELSE                                                         COMQSVR    129   A
COMQSVR            MI     "B=1",*+4S15  IF ("B=1") IS BAD (SHOULD BE EQUAL TO 1) COMQSVR    130   A
COMQSVR            S"B=1" "B=1"-1                                               COMQSVR    131   A
COMQSVR            ZR     "B=1",SVR2  IF ("B=1") = 1                           COMQSVR    132   A
COMQSVR            S"B=1" "B=1"+1     RESTORE ("B=1")                           COMQSVR    133   A
COMQSVR            EQ     *+4S15                                               COMQSVR    134   A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMQSVR

```
COMQSVR                                                                  COMQSVR      135     A
COMQSVR   SVR2     BSS    0                                              COMQSVR      136     A
COMQSVR            IFC    NE,/"B=1"/B1/,1                                COMQSVR      137     A
COMQSVR            S"B=1" B1-B0       SAVE (B1)                          COMQSVR      138     A
COMQSVR   B.EQ.1   ENDIF                                                 COMQSVR      139     A
COMQSVR                                                                  COMQSVR      140     A
COMQSVR  *         SAVE   A - 4,5,6,7                                    COMQSVR      141     A
COMQSVR  *                X - 4,5,6,7                                    COMQSVR      142     A
COMQSVR                                                                  COMQSVR      143     A
COMQSVR   .SVR1    IF     -DEF,#SVR   IF USING LOCAL REGISTER SAVE AREA  COMQSVR      144     A
COMQSVR            SB1    A7-B0                                          COMQSVR      145     A
COMQSVR            SA7    SVREG+SVX+7                                    COMQSVR      146     A
COMQSVR            SX7    B1-B0                                          COMQSVR      147     A
COMQSVR            SA7    SVREG+SVA+7                                    COMQSVR      148     A
COMQSVR            SB1    1                                              COMQSVR      149     A
COMQSVR            SX7    A6-B0                                          COMQSVR      150     A
COMQSVR            SA6    SVREG+SVX+6                                    COMQSVR      151     A
COMQSVR            SA7    A7-B1                                          COMQSVR      152     A
COMQSVR            BX6    X5                                             COMQSVR      153     A
COMQSVR            SX7    A5-B0                                          COMQSVR      154     A
COMQSVR            SA6    A6-B1                                          COMQSVR      155     A
COMQSVR            SA7    A7-B1                                          COMQSVR      156     A
COMQSVR            BX6    X4                                             COMQSVR      157     A
COMQSVR            SX7    A4-B0                                          COMQSVR      158     A
COMQSVR            SA6    A6-B1                                          COMQSVR      159     A
COMQSVR            SA7    A7-B1                                          COMQSVR      160     A
COMQSVR            SX5    SVREG       (X5) = FWA OF SAVED REGISTER AREA  COMQSVR      161     A
COMQSVR                                                                  COMQSVR      162     A
COMQSVR   .SVR1    ELSE               IF USING USER DEFINED SAVE AREA   COMQSVR      163     A
COMQSVR            SB1    A7-B0       SAVE (A7) ** TEMPORARY **          COMQSVR      164     A
COMQSVR            SA7    SVX7        SAVE (X7) ** TEMPORARY **          COMQSVR      165     A
COMQSVR            SX7    A5-B0                                          COMQSVR      166     A
COMQSVR            SA7    SVA5        SAVE (A5) ** TEMPORARY **          COMQSVR      167     A
COMQSVR            BX7    X5                                             COMQSVR      168     A
COMQSVR            SA7    SVX5        SAVE (X5) ** TEMPORARY **          COMQSVR      169     A
COMQSVR                                                                  COMQSVR      170     A
COMQSVR            SA5    SVR=        (X5) = 30/EQ CALLER+1, 30/0        COMQSVR      171     A
COMQSVR            LX5    30                                             COMQSVR      172     A
COMQSVR            SA5    X5-1        (X5) = 30/RJ SVR=, 30/FWA OF RSA   COMQSVR      173     A
COMQSVR            SX7    B1-B0                                          COMQSVR      174     A
COMQSVR            SB1    1                                              COMQSVR      175     A
COMQSVR            SA7    X5+SVA+7                                       COMQSVR      176     A
COMQSVR            SX7    A6-B0                                          COMQSVR      177     A
COMQSVR            SA6    X5+SVX+6                                       COMQSVR      178     A
COMQSVR            SA7    A7-B1                                          COMQSVR      179     A
COMQSVR            BX6    X4                                             COMQSVR      180     A
COMQSVR            SX7    A4-B0                                          COMQSVR      181     A
COMQSVR            SA6    A6-2                                           COMQSVR      182     A
COMQSVR            SA7    A7-2                                           COMQSVR      183     A
COMQSVR   .SVR1    ENDIF                                                 COMQSVR      184     A
COMQSVR                                                                  COMQSVR      185     A
COMQSVR  *         SAVE   A - 0,1,2,3                                    COMQSVR      186     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMQSVR

```
COMQSVR  *                X - 0,1,2,3                                          COMQSVR      187     A
COMQSVR                                                                        COMQSVR      188     A
COMQSVR   SV=AX    ECHO   ,N=(3,2,1,0)                                         COMQSVR      189     A
COMQSVR            BX6    X.N                                                  COMQSVR      190     A
COMQSVR            SX7    A.N-B0                                               COMQSVR      191     A
COMQSVR            SA6    A6-B1                                                COMQSVR      192     A
COMQSVR            SA7    A7-B1                                                COMQSVR      193     A
COMQSVR   SV=AX    ENDD                                                        COMQSVR      194     A
COMQSVR                                                                        COMQSVR      195     A
COMQSVR  *         IF USING USER DEFINED SAVE AREA, SAVE (X7),(A5),(X5) FINALLY.  COMQSVR  196     A
COMQSVR                                                                        COMQSVR      197     A
COMQSVR   .SVR2    IF     DEF,#SVR                                             COMQSVR      198     A
COMQSVR            SA2    SVX7        (X2) = SAVED (X7)                        COMQSVR      199     A
COMQSVR            SA3    A2+B1       (X3) = SAVED (A5)                        COMQSVR      200     A
COMQSVR            SA4    A3+B1       (X4) = SAVED (X5)                        COMQSVR      201     A
COMQSVR            BX6    X2                                                   COMQSVR      202     A
COMQSVR            LX7    X3                                                   COMQSVR      203     A
COMQSVR            SA6    X5+SVX+7                                             COMQSVR      204     A
COMQSVR            SA7    X5+SVA+5                                             COMQSVR      205     A
COMQSVR            BX6    X4                                                   COMQSVR      206     A
COMQSVR            SA6    X5+SVX+5                                             COMQSVR      207     A
COMQSVR   .SVR2    ENDIF                                                       COMQSVR      208     A
COMQSVR                                                                        COMQSVR      209     A
COMQSVR  *         SAVE   B - 0,2,3,4,5,6,7                                   COMQSVR      210     A
COMQSVR                                                                        COMQSVR      211     A
COMQSVR            BX6    X6-X6                                                COMQSVR      212     A
COMQSVR            SA6    X5+SVB+0                                             COMQSVR      213     A
COMQSVR                                                                        COMQSVR      214     A
COMQSVR   B=1      IFC    NE,/"B=1"/B0/                                        COMQSVR      215     A
COMQSVR            SX7    "B=1"-B0                                             COMQSVR      216     A
COMQSVR            S"B=1" B1          RESET ("B=1") = 1                       COMQSVR      217     A
COMQSVR                                                                        COMQSVR      218     A
COMQSVR   B=1      ELSE                                                        COMQSVR      219     A
COMQSVR            SX7    B1                                                   COMQSVR      220     A
COMQSVR   B=1      ENDIF                                                       COMQSVR      221     A
COMQSVR                                                                        COMQSVR      222     A
COMQSVR            SA7    A6+B1                                                COMQSVR      223     A
COMQSVR                                                                        COMQSVR      224     A
COMQSVR   SV=B     ECHO   ,U=(2,4,6),L=(3,5,7)                                 COMQSVR      225     A
COMQSVR            SX6    B.U-B0                                               COMQSVR      226     A
COMQSVR            SX7    B.L-B0                                               COMQSVR      227     A
COMQSVR            SA6    A7+B1                                                COMQSVR      228     A
COMQSVR            SA7    A6+B1                                                COMQSVR      229     A
COMQSVR   SV=B     ENDD                                                        COMQSVR      230     A
COMQSVR                                                                        COMQSVR      231     A
COMQSVR  *         SAVE (B1)                                                   COMQSVR      232     A
COMQSVR                                                                        COMQSVR      233     A
COMQSVR   B.NE.1   IFC    EQ,/"B=1"/B0/                                        COMQSVR      234     A
COMQSVR            SA5    SVR3        (X5) = (B1)=1/(B1).NE.1 FLAG             COMQSVR      235     A
COMQSVR            SX4    0100B       (X4) = *RJ* OP CODE                      COMQSVR      236     A
COMQSVR            SX3    B1                                                   COMQSVR      237     A
COMQSVR            LX3    42-0-1                                               COMQSVR      238     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMQSVR

```
COMQSVR            SX6    A5                                                    COMQSVR    239    A
COMQSVR            LX4    29-11                                                 COMQSVR    240    A
COMQSVR            BX6    X4+X6       (X6) = 30/0, 30/RJ *                      COMQSVR    241    A
COMQSVR            LX6    30                                                    COMQSVR    242    A
COMQSVR            SA6    A5          RESTORE .RJ *.                            COMQSVR    243    A
COMQSVR            LX5    59-56                                                 COMQSVR    244    A
COMQSVR            BX7    X7-X7                                                 COMQSVR    245    A
COMQSVR            PL     X5,SVR=     IF (B1) = 1, NO NEED TO ASSEMBLE (B1)     COMQSVR    246    A
COMQSVR            SB2    B1+B1                                                 COMQSVR    247    A
COMQSVR                                                                         COMQSVR    248    A
COMQSVR  *         ASSEMBLE ORIGINAL (B1) INTO (X7).                            COMQSVR    249    A
COMQSVR                                                                         COMQSVR    250    A
COMQSVR   SVR5     SA5    A5-B2                                                 COMQSVR    251    A
COMQSVR            IX3    X3+X3                                                 COMQSVR    252    A
COMQSVR            SX6    A5                                                    COMQSVR    253    A
COMQSVR            LX5    59-56                                                 COMQSVR    254    A
COMQSVR            BX6    X4+X6                                                 COMQSVR    255    A
COMQSVR            PL     X5,SVR6     IF BIT WAS CLEAR                          COMQSVR    256    A
COMQSVR            BX7    X7+X3                                                 COMQSVR    257    A
COMQSVR                                                                         COMQSVR    258    A
COMQSVR   SVR6     LX6    59-29                                                 COMQSVR    259    A
COMQSVR            SA6    A5                                                    COMQSVR    260    A
COMQSVR            PL     X3,SVR5     IF MORE BITS TO ASSEMBLE                  COMQSVR    261    A
COMQSVR                                                                         COMQSVR    262    A
COMQSVR  *         HERE FOR FINAL SAVE OF (B1).                                 COMQSVR    263    A
COMQSVR                                                                         COMQSVR    264    A
COMQSVR            AX7    42          SIGN EXTEND                               COMQSVR    265    A
COMQSVR            SA7    A7-7+1                                                COMQSVR    266    A
COMQSVR   B.NE.1   ENDIF                                                        COMQSVR    267    A
COMQSVR                                                                         COMQSVR    268    A
COMQSVR            EQ     SVR=        EXIT...                                   COMQSVR    269    A
COMQSVR   RSR=     SPACE  4,8                                                   COMQSVR    270    A
COMQSVR  **        RSR - RESTORE ALL REGISTERS.                                 COMQSVR    271    A
COMQSVR  *                                                                      COMQSVR    272    A
COMQSVR  *                                                                      COMQSVR    273    A
COMQSVR  *         ENTRY  IF *#SVR* NOT DEFINED-- NONE                          COMQSVR    274    A
COMQSVR  *                                                                      COMQSVR    275    A
COMQSVR  *                IF *#SVR* DEFINED-- (X1) = FWA OF REGISTER SAVE AREA  COMQSVR    276    A
COMQSVR  *                                                                      COMQSVR    277    A
COMQSVR  *         EXIT   ALL REGISTERS SET TO THE CONTENTS OF REGISTER SAVE AREA COMQSVR  278    A
COMQSVR  *                                                                      COMQSVR    279    A
COMQSVR  *         USES   ALL                                                   COMQSVR    280    A
COMQSVR  *                                                                      COMQSVR    281    A
COMQSVR  *         CALLS  NONE                                                  COMQSVR    282    A
COMQSVR                                                                         COMQSVR    283    A
COMQSVR                                                                         COMQSVR    284    A
COMQSVR   RSR=     EQ     *+4S15      ** ENTRY/EXIT **                          COMQSVR    285    A
COMQSVR            SB1    1                                                     COMQSVR    286    A
COMQSVR                                                                         COMQSVR    287    A
COMQSVR  *         SET UP FINAL -B- REGISTERS RESTORE AT (RSR4) ET SEQ.         COMQSVR    288    A
COMQSVR                                                                         COMQSVR    289    A
COMQSVR   .RSR     IF     DEF,#SVR    IF USING USER DEFINED REGISTER SAVE AREA  COMQSVR    290    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMQSVR

```
COMQSVR            SB7    X1+          (B7) = FWA OF REGISTER SAVE AREA          COMQSVR    291    A
COMQSVR                                                                           COMQSVR    292    A
COMQSVR   .RSR     ELSE                IF USING LOCAL REGISTER SAVE AREA         COMQSVR    293    A
COMQSVR            SB7    SVREG                                                   COMQSVR    294    A
COMQSVR   .RSR     ENDIF                                                           COMQSVR    295    A
COMQSVR                                                                           COMQSVR    296    A
COMQSVR            MX4    -18                                                     COMQSVR    297    A
COMQSVR            SA1    B7+SVB+7     (X1) = SAVED (B7)                          COMQSVR    298    A
COMQSVR            BX5    X4                                                      COMQSVR    299    A
COMQSVR            LX4    30                                                      COMQSVR    300    A
COMQSVR            SB4    4                                                       COMQSVR    301    A
COMQSVR            BX0    X4*X5        (X0) = 77770000007777000000B               COMQSVR    302    A
COMQSVR                                                                           COMQSVR    303    A
COMQSVR   RSR2     SA2    A1-B1        (X2) = SAVED (B.N-1)                       COMQSVR    304    A
COMQSVR            SA3    RSR4+B4-1                                               COMQSVR    305    A
COMQSVR            BX1    -X5*X1                                                  COMQSVR    306    A
COMQSVR            SB4    B4-B1                                                   COMQSVR    307    A
COMQSVR            BX2    -X5*X2                                                  COMQSVR    308    A
COMQSVR            LX1    30                                                      COMQSVR    309    A
COMQSVR            BX3    X0*X3                                                   COMQSVR    310    A
COMQSVR            IX6    X1+X2                                                   COMQSVR    311    A
COMQSVR            BX7    X6+X3        (X7) = 30/SB.N B7+K, 30/SB.N-1 B7+K        COMQSVR    312    A
COMQSVR            SA7    A3                                                      COMQSVR    313    A
COMQSVR            SA1    A2-B1        (X1) = SAVED (B.N)                         COMQSVR    314    A
COMQSVR            GT     B4,B0,RSR2   IF STILL MORE -B- REGS TO GO...            COMQSVR    315    A
COMQSVR                                                                           COMQSVR    316    A
COMQSVR            SA2    B7+SVA+6     (X2) = SAVED (A6)                          COMQSVR    317    A
COMQSVR            RJ     *+1          VOID INSTRUCTION STACK                     CCGA090      6    A
COMQSVR    +       EQ     *                                                       CCGA090      7    A
COMQSVR                                                                           COMQSVR    318    A
COMQSVR   *        THE MOST LIKELY CANDIDATE FOR THE DANGER DISCUSSED IN THE NEXT COMQSVR    319    A
COMQSVR   *          PARAGRAPH IS THE SYSTEM COMMUNICATION CELL (RA.MTR).          COMQSVR    320    A
COMQSVR   *          ACCORDINGLY, WE WAIT FOR IT TO CLEAR BEFORE PROCEEDING.       COMQSVR    321    A
COMQSVR                                                                           COMQSVR    322    A
COMQSVR    RSR3    SA1    B1                                                      COMQSVR    323    A
COMQSVR            NZ     X1,RSR3      IF (RA.MTR) NOT CLEAR                      COMQSVR    324    A
COMQSVR                                                                           COMQSVR    325    A
COMQSVR   *        A SLIGHT BIT OF CARE IS SPENT ON RESTORATION OF (A6) AND (A7). COMQSVR    326    A
COMQSVR   *          WE WISH TO AVOID HAVING THE CONTENTS OF THE WORD THAT THE     COMQSVR    327    A
COMQSVR   *          STORE REGISTER IS POINTING TO TO GET CHANGED (I.E. BY A PP)   COMQSVR    328    A
COMQSVR   *          WHILE WE ARE RESTORING (A6) OR (A7). ABOUT THE ONLY THING     COMQSVR    329    A
COMQSVR   *          THAT CAN BE DONE TO MINIMIZE THIS POSSIBILITY IS TO GET THE   COMQSVR    330    A
COMQSVR   *          LOAD AND THE STORE AS CLOSE TOGETHER AS POSSIBLE. BY KEEPING  COMQSVR    331    A
COMQSVR   *          THEM IN THE SAME WORD, WE HAVE AT LEAST INSURED THAT WE WILL  COMQSVR    332    A
COMQSVR   *          NOT BE EXCHANGED AFTER THE LOAD BUT BEFORE THE STORE.         COMQSVR    333    A
COMQSVR   *          BARRING DELAYS IN THE HOPPER, THIS SHOULD REDUCE THE DANGER   COMQSVR    334    A
COMQSVR   *          PERIOD TO A FEW MINOR CYCLES. SUGGESTIONS ARE WELCOME.        COMQSVR    335    A
COMQSVR                                                                           COMQSVR    336    A
COMQSVR            SA3    A2+B1        (X3) = SAVED (A7)                          COMQSVR    337    A
COMQSVR                                                                           COMQSVR    338    A
COMQSVR    +       SA4    X2-0                                                    COMQSVR    339    A
COMQSVR            BX6    X4                                                      COMQSVR    340    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMQSVR

```
COMQSVR            SA6    A4-B0       RESTORE (A6)                                    COMQSVR      341     A
COMQSVR                                                                               COMQSVR      342     A
COMQSVR   +        SA5    X3-0                                                        COMQSVR      343     A
COMQSVR            BX7    X5                                                          COMQSVR      344     A
COMQSVR            SA7    A5-B0       RESTORE (A7)                                    COMQSVR      345     A
COMQSVR                                                                               COMQSVR      346     A
COMQSVR            SA1    B7+SVX+6                                                    COMQSVR      347     A
COMQSVR            SA4    A1+B1                                                       COMQSVR      348     A
COMQSVR            BX6    X1          RESTORE (X6)                                    COMQSVR      349     A
COMQSVR            LX7    X4          RESTORE (X7)                                    COMQSVR      350     A
COMQSVR            BX0    -X0-X0                                                      COMQSVR      351     A
COMQSVR            SB6    X0-0        (B6) = -0                                       COMQSVR      352     A
COMQSVR                                                                               COMQSVR      353     A
COMQSVR   *        RESTORE (A0),(X0).                                                 COMQSVR      354     A
COMQSVR                                                                               COMQSVR      355     A
COMQSVR            SA4    B7+SVA+0                                                    COMQSVR      356     A
COMQSVR            SA5    B7+SVX+0                                                    COMQSVR      357     A
COMQSVR            SA0    X4+B6                                                       COMQSVR      358     A
COMQSVR            BX0    X5                                                          COMQSVR      359     A
COMQSVR                                                                               COMQSVR      360     A
COMQSVR   *        RESTORE A - 1,2,3,4                                                COMQSVR      361     A
COMQSVR   *                 X - 1,2,3,4                                               COMQSVR      362     A
COMQSVR                                                                               COMQSVR      363     A
COMQSVR            ECHO   4,N=(1,2,3,4)                                               COMQSVR      364     A
COMQSVR            SA4    A4+B1       (X4) = SAVED (A.N)                              COMQSVR      365     A
COMQSVR            SA5    A5+B1       (X5) = SAVED (X.N)                              COMQSVR      366     A
COMQSVR            SA.N   X4+B6                                                       COMQSVR      367     A
COMQSVR            BX.N   X5                                                          COMQSVR      368     A
COMQSVR                                                                               COMQSVR      369     A
COMQSVR   *        RESTORE (A5),(X5)-- THE ORIGINAL (X5) ARE BROKEN APART INTO        COMQSVR      370     A
COMQSVR   *          (B1) THRU (B5) AS DISPLAYED BELOW. THIS ALLOWS US TO RESTORE     COMQSVR      371     A
COMQSVR   *          (A5).                                                            COMQSVR      372     A
COMQSVR   *          DISASSEMBLY IS FROM LEFT TO RIGHT, TOP TO BOTTOM.                COMQSVR      373     A
COMQSVR   *          REASSEMBLY IS FROM RIGHT TO LEFT, BOTTOM TO TOP.                 COMQSVR      374     A
COMQSVR   *        5          4          3          2          1               55     COMQSVR      375     A
COMQSVR   *        9          8          7          6          5              098     COMQSVR      376     A
COMQSVR   *        ............................................................       COMQSVR      377     A
COMQSVR   *        ++++++++++++                                                       COMQSVR      378     A
COMQSVR   *            (B1)   ++++++++++++                                            COMQSVR      379     A
COMQSVR   *                       (B2)   ++++++++++++                                 COMQSVR      380     A
COMQSVR   *                                   (B3)   ++++++++++++                     COMQSVR      381     A
COMQSVR   *                                              (B4)   ++++++++++++++++++    COMQSVR      382     A
COMQSVR   *        ++                                                  (B5)           COMQSVR      383     A
COMQSVR                                                                               COMQSVR      384     A
COMQSVR            SA5    A5+B1       (X5) = SAVED (X5)                               COMQSVR      385     A
COMQSVR                                                                               COMQSVR      386     A
COMQSVR            ECHO   2,N=(1,2,3)                                                 COMQSVR      387     A
COMQSVR            UX5    X5,B.N                                                      COMQSVR      388     A
COMQSVR            LX5    11                                                          COMQSVR      389     A
COMQSVR                                                                               COMQSVR      390     A
COMQSVR            UX5    X5,B4                                                       COMQSVR      391     A
COMQSVR            LX5    11+18                                                       COMQSVR      392     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMQSVR

```
COMQSVR            SB5    X5+B6                                                    COMQSVR      393     A
COMQSVR                                                                             COMQSVR      394     A
COMQSVR            SA5    B7+SVA+5    (X5) = SAVED (A5)                            COMQSVR      395     A
COMQSVR            SA5    X5+B6       RESTORE (A5)                                 COMQSVR      396     A
COMQSVR                                                                             COMQSVR      397     A
COMQSVR  *         REASSEMBLE (X5).                                                COMQSVR      398     A
COMQSVR                                                                             COMQSVR      399     A
COMQSVR            SX5    B5+B6                                                    COMQSVR      400     A
COMQSVR            LX5    60-11-18                                                 COMQSVR      401     A
COMQSVR            PX5    X5,B4                                                    COMQSVR      402     A
COMQSVR                                                                             COMQSVR      403     A
COMQSVR            ECHO   2,N=(3,2,1)                                              COMQSVR      404     A
COMQSVR            LX5    -11                                                      COMQSVR      405     A
COMQSVR            PX5    X5,B.N                                                   COMQSVR      406     A
COMQSVR                                                                             COMQSVR      407     A
COMQSVR  *         RESTORE B - 0,1,2,3,4,5,6,7                                     COMQSVR      408     A
COMQSVR                                                                             COMQSVR      409     A
COMQSVR   RSR4     BSS    0                                                        COMQSVR      410     A
COMQSVR            ECHO   2,U=(1,3,5,7),L=(0,2,4,6)                                COMQSVR      411     A
COMQSVR            SB.U   B6+*                                                     COMQSVR      412     A
COMQSVR            SB.L   B6+*                                                     COMQSVR      413     A
COMQSVR                                                                             COMQSVR      414     A
COMQSVR            EQ     RSR=        EXIT...                                      COMQSVR      415     A
COMQSVR   SVR=     SPACE  4,8                                                      COMQSVR      416     A
COMQSVR            BASE   *                                                        COMQSVR      417     A
COMQSVR                                                                             COMQSVR      418     A
COMQSVR   #SVR     IF     -DEF,#SVR                                                COMQSVR      419     A
COMQSVR   SV=B     =      SVREG+SVB                                                COMQSVR      420     A
COMQSVR   SV=A     =      SVREG+SVA                                                COMQSVR      421     A
COMQSVR   SV=X     =      SVREG+SVX                                                COMQSVR      422     A
COMQSVR   #SVR     ENDIF                                                           COMQSVR      423     A
COMQSVR                                                                             COMQSVR      424     A
COMQSVR   QUAL$    IF     -DEF,QUAL$                                               COMQSVR      425     A
COMQSVR            QUAL   *                                                        COMQSVR      426     A
COMQSVR   SVR=     =      /COMQSVR/SVR=                                            COMQSVR      427     A
COMQSVR   RSR=     =      /COMQSVR/RSR=                                            COMQSVR      428     A
COMQSVR                                                                             COMQSVR      429     A
COMQSVR   #SVR     IF     -DEF,#SVR                                                COMQSVR      430     A
COMQSVR   SV=B     =      /COMQSVR/SV=B                                            COMQSVR      431     A
COMQSVR   SV=A     =      /COMQSVR/SV=A                                            COMQSVR      432     A
COMQSVR   SV=X     =      /COMQSVR/SV=X                                            COMQSVR      433     A
COMQSVR   #SVR     ENDIF                                                           COMQSVR      434     A
COMQSVR   QUAL$    ENDIF                                                           COMQSVR      435     A
COMQSVR            ENDX                                                            COMQSVR      436     A
```

SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - COMQSVR

IDENTIFIER	TOTAL	ACTIVE
COMQSVR	436	436
CCGA090	2	2

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMQSVR

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMSIDP

```
COMSIDP  *COMDECK COMSIDP                                                  COMSIDP        1     A
COMSIDP   COMSIDP  TITLE  COMSIDP - INTERACTIVE DEBUG PACKAGE INTERFACE TEXT.  COMSIDP        2     A
COMSIDP   COMSIDP  SPACE  4,8                                              COMSIDP        3     A
COMSIDP            BASE   D                                                COMSIDP        4     A
COMSIDP   COMSIDP  SPACE  4,8                                              COMSIDP        5     A
COMSIDP  ***       COMSIDP - INTERACTIVE DEBUG PACKAGE INTERFACE TEXT.     COMSIDP        6     A
COMSIDP  *         75/11/01           P CLARE                              COMSIDP        7     A
COMSIDP  *         77/05/12           P CLARE                              COMSIDP        8     A
COMSIDP   COMSIDP  SPACE  4,8                                              COMSIDP        9     A
COMSIDP  ***       COMSIDP - INTERACTIVE DEBUG PACKAGE INTERFACE TEXT.     COMSIDP       10     A
COMSIDP  *                                                                 COMSIDP       11     A
COMSIDP  *                                                                 COMSIDP       12     A
COMSIDP  *         *COMSIDP* CONTAINS THE MACRO AND SYMBOL DEFINITIONS USED COMSIDP       13     A
COMSIDP  *         BY IDP AND ANY USER ROUTINES THAT ARE INTERFACING TO IT. COMSIDP       14     A
COMSIDP   .OS      SPACE  4,10                                             COMSIDP       15     A
COMSIDP  **        DEFINE DEFAULT OPERATING SYSTEM.                        COMSIDP       16     A
COMSIDP                                                                    COMSIDP       17     A
COMSIDP            IF     -DEF,.OS,1                                       COMSIDP       18     A
COMSIDP   .OS      =      3           SCOPE 3                              COMSIDP       19     A
COMSIDP   CEQU     SPACE  4,10                                             COMSIDP       20     A
COMSIDP  **        CEQU - CONDITIONAL EQU.                                 COMSIDP       21     A
COMSIDP  *                                                                 COMSIDP       22     A
COMSIDP  *                                                                 COMSIDP       23     A
COMSIDP  * A       CEQU   B                                                COMSIDP       24     A
COMSIDP  *                                                                 COMSIDP       25     A
COMSIDP  *         ENTRY  A = SYMBOL NAME TO EQU TO *B*, ONLY IF *A* NOT   COMSIDP       26     A
COMSIDP  *                     ALREADY DEFINED.                            COMSIDP       27     A
COMSIDP  *                B  = NEW VALUE FOR *A*, IF *A* NOT ALREADY DEFINED. COMSIDP       28     A
COMSIDP  *                                                                 COMSIDP       29     A
COMSIDP  *         EXIT   NONE                                             COMSIDP       30     A
COMSIDP  *                                                                 COMSIDP       31     A
COMSIDP  *         USES   NONE                                             COMSIDP       32     A
COMSIDP  *                                                                 COMSIDP       33     A
COMSIDP  *         CALLS  NONE                                             COMSIDP       34     A
COMSIDP                                                                    COMSIDP       35     A
COMSIDP                                                                    COMSIDP       36     A
COMSIDP            PURGMAC CEQU                                            COMSIDP       37     A
COMSIDP                                                                    COMSIDP       38     A
COMSIDP            MACRO  CEQU,AA,BB                                       COMSIDP       39     A
COMSIDP    IF -DEF,AA,1                                                    COMSIDP       40     A
COMSIDP   AA = BB                                                          COMSIDP       41     A
COMSIDP   CEQU ENDM                                                        COMSIDP       42     A
COMSIDP   DEFTOT   SPACE  4,10                                             COMSIDP       43     A
COMSIDP  **        DEFTOT - DEFINE TOKEN TYPE.                             COMSIDP       44     A
COMSIDP  *                                                                 COMSIDP       45     A
COMSIDP  *                                                                 COMSIDP       46     A
```

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMSIDP

```
COMSIDP  * TOT     DEFTOT                                                  COMSIDP     47    A
COMSIDP  *                                                                 COMSIDP     48    A
COMSIDP  *         ENTRY  TOT = TOKEN TYPE TO DEFINE.  NOTE THAT *TOT* WILL  COMSIDP  49    A
COMSIDP  *                      BE PREFIXED WITH "O.".                     COMSIDP     50    A
COMSIDP  *                                                                 COMSIDP     51    A
COMSIDP  *         EXIT   NONE                                             COMSIDP     52    A
COMSIDP  *                                                                 COMSIDP     53    A
COMSIDP  *         USES   NONE                                             COMSIDP     54    A
COMSIDP  *                                                                 COMSIDP     55    A
COMSIDP  *         CALLS  NONE                                             COMSIDP     56    A
COMSIDP                                                                    COMSIDP     57    A
COMSIDP                                                                    COMSIDP     58    A
COMSIDP            PURGMAC DEFTOT                                          COMSIDP     59    A
COMSIDP                                                                    COMSIDP     60    A
COMSIDP            MACRO  DEFTOT,TOT                                       COMSIDP     61    A
COMSIDP  *                                                                 COMSIDP     62    A
COMSIDP    IF -DEF,'?IDSTOT,1                                              COMSIDP     63    A
COMSIDP   '?IDSTOT SET -1                                                  COMSIDP     64    A
COMSIDP  *                                                                 COMSIDP     65    A
COMSIDP   '?IDSTOT SET '?IDSTOT+1                                          COMSIDP     66    A
COMSIDP  *                                                                 COMSIDP     67    A
COMSIDP   O.TOT = '?IDSTOT                                                 COMSIDP     68    A
COMSIDP  *                                                                 COMSIDP     69    A
COMSIDP   DEFTOT   ENDM                                                    COMSIDP     70    A
COMSIDP   ERROR    SPACE  4,10                                             COMSIDP     71    A
COMSIDP  **        ERROR - DEFINE IDP ERROR MESSAGE.                       COMSIDP     72    A
COMSIDP  *                                                                 COMSIDP     73    A
COMSIDP  *                                                                 COMSIDP     74    A
COMSIDP  * LOC     ERROR  MSG,JMP                                          COMSIDP     75    A
COMSIDP  *                                                                 COMSIDP     76    A
COMSIDP  *         ENTRY  LOC = LOCATION FIELD TO JUMP TO FOR THIS ERROR.  COMSIDP     77    A
COMSIDP  *                MSG = ERROR MSG TEXT.                            COMSIDP     78    A
COMSIDP  *                JMP = ADDR TO JUMP TO FOR THIS ERROR,            COMSIDP     79    A
COMSIDP  *                      (DEFAULT = *IDP=ERR*).                     COMSIDP     80    A
COMSIDP  *                                                                 COMSIDP     81    A
COMSIDP  *         EXIT   NONE                                             COMSIDP     82    A
COMSIDP  *                                                                 COMSIDP     83    A
COMSIDP  *         USES   NONE                                             COMSIDP     84    A
COMSIDP  *                                                                 COMSIDP     85    A
COMSIDP  *         CALLS  NONE                                             COMSIDP     86    A
COMSIDP                                                                    COMSIDP     87    A
COMSIDP                                                                    COMSIDP     88    A
COMSIDP            MACRO  ERROR,LOC,MSG,JMP                                COMSIDP     89    A
COMSIDP  *                                                                 COMSIDP     90    A
COMSIDP   '?ID#002 IFNE OI.ERR,0                                           COMSIDP     91    A
COMSIDP  *                                                                 COMSIDP     92    A
COMSIDP   LOC BSS 0                                                        COMSIDP     93    A
COMSIDP  *                                                                 COMSIDP     94    A
COMSIDP    IF -DEF,'?IDSERR,1                                              COMSIDP     95    A
COMSIDP   '?IDSERR SET 0                                                   COMSIDP     96    A
COMSIDP  *                                                                 COMSIDP     97    A
COMSIDP   '?IDSERR SET '?IDSERR+1                                          COMSIDP     98    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMSIDP

```
COMSIDP  *                                                          COMSIDP      99    A
COMSIDP   '?IDCS01 MICRO 1,,# MSG_#                                 COMSIDP     100    A
COMSIDP   '?IDS001 MICCNT '?IDCS01                                  COMSIDP     101    A
COMSIDP    IFGT '?IDS001,80D,1                                      COMSIDP     102    A
COMSIDP   P ERR ERR MESSAGE TOO LONG.                               COMSIDP     103    A
COMSIDP  *                                                          COMSIDP     104    A
COMSIDP   '?ID#001 IFEQ OI.ERR,2                                    COMSIDP     105    A
COMSIDP    SX6 =C# MSG_#                                            COMSIDP     106    A
COMSIDP   '?ID#001 ELSE                                             COMSIDP     107    A
COMSIDP    SX6 '?IDSERR                                             COMSIDP     108    A
COMSIDP   '?ID#001 ENDIF                                            COMSIDP     109    A
COMSIDP  *                                                          COMSIDP     110    A
COMSIDP   '?ID#002 ELSE                                             COMSIDP     111    A
COMSIDP   LOC = IDP=ER                                              COMSIDP     112    A
COMSIDP   '?ID#002 ENDIF                                            COMSIDP     113    A
COMSIDP  *                                                          COMSIDP     114    A
COMSIDP    EQ JMP IDP=ERR                                           COMSIDP     115    A
COMSIDP  *                                                          COMSIDP     116    A
COMSIDP   ERROR    ENDM                                             COMSIDP     117    A
COMSIDP   KEYW     SPACE  4,10                                      COMSIDP     118    A
COMSIDP  **        KEYW - *IDP* KEYWORD TABLE GENERATOR.            COMSIDP     119    A
COMSIDP  *                                                          COMSIDP     120    A
COMSIDP  *                                                          COMSIDP     121    A
COMSIDP  *         MACRO USED TO GENERATE THE *IDP* COMMAND KEYWORD TABLE.  COMSIDP 122  A
COMSIDP  *                                                          COMSIDP     123    A
COMSIDP  * LAB     KEYW   KEY,XAT,ATTR                              COMSIDP     124    A
COMSIDP  *                                                          COMSIDP     125    A
COMSIDP  *         ENTRY  LAB  = CHARS USED FOR GENERATING USER-SPECIFIED  COMSIDP 126  A
COMSIDP  *                       ADDR REFS FOR THIS KEYWORD.  THAT IS, THE  COMSIDP 127  A
COMSIDP  *                       *KEYW* MACRO WILL CONCATENATE PREDETERMINED  COMSIDP 128  A
COMSIDP  *                       CHARACTERS TO *LAB* TO GENERATE THE FOLLOWING  COMSIDP 129  A
COMSIDP  *                       ADDRESSES --                       COMSIDP     130    A
COMSIDP  *                                                          COMSIDP     131    A
COMSIDP  *                         ST=LAB  = ADDR OF STMT PROCESSOR FOR THIS  COMSIDP 132  A
COMSIDP  *                                   KEYWORD.               COMSIDP     133    A
COMSIDP  *                         ST.LAB  = ADDR OF WORD 1 OF THIS *KEYW* ENTRY.  COMSIDP 134  A
COMSIDP  *                         LIP=LAB = ADDR OF *LIST* PROCESSOR FOR THIS  COMSIDP 135  A
COMSIDP  *                                   KEYWORD.               COMSIDP     136    A
COMSIDP  *                                                          COMSIDP     137    A
COMSIDP  *                 KEY  = CHARS THAT CONSTITUTE THIS KEYWORD AND *HELP*  COMSIDP 138  A
COMSIDP  *                       STRING.  FORMAT IS AS FOLLOWS --   COMSIDP     139    A
COMSIDP  *                                                          COMSIDP     140    A
COMSIDP  *                         (BREAK FWA,LWA,LEN,LL,UL,INC)    COMSIDP     141    A
COMSIDP  *                                                          COMSIDP     142    A
COMSIDP  *                       NOTE THAT ACTUAL KEYWORD AND *HELP* STRING  COMSIDP 143  A
COMSIDP  *                       **MUST** BE SEPARATED BY A BLANK (55B) CHAR.  COMSIDP 144  A
COMSIDP  *                                                          COMSIDP     145    A
COMSIDP  *                       NOTE ALSO THAT *HELP* STRING IS OPTIONAL.  COMSIDP 146  A
COMSIDP  *                                                          COMSIDP     147    A
COMSIDP  *                 XAT  = EXCLUSIVE ATTRIBUTE FOR THIS KEYWORD.  COMSIDP 148  A
COMSIDP  *                                                          COMSIDP     149    A
COMSIDP  *                 ATTR = NON-EXCLUSIVE ATTRIBUTES FOR THIS KEYWORD.  COMSIDP 150  A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMSIDP

```
COMSIDP  *                      NULL IF NONE.  (CURRENTLY UNUSED FIELD).  COMSIDP  151  A
COMSIDP  *                                                                COMSIDP  152  A
COMSIDP  *         EXIT   NONE                                            COMSIDP  153  A
COMSIDP  *                                                                COMSIDP  154  A
COMSIDP  *         USES   NONE                                            COMSIDP  155  A
COMSIDP  *                                                                COMSIDP  156  A
COMSIDP  *         CALLS  '?IDMKWL                                        COMSIDP  157  A
COMSIDP                                                                   COMSIDP  158  A
COMSIDP                                                                   COMSIDP  159  A
COMSIDP            PURGMAC  KEYW                                          COMSIDP  160  A
COMSIDP                                                                   COMSIDP  161  A
COMSIDP            MACRO  KEYW,LAB,KEY,XA,ATTR                            COMSIDP  162  A
COMSIDP            LOCAL  '?ID#001                                        COMSIDP  163  A
COMSIDP  *                                                                COMSIDP  164  A
COMSIDP   '?IDCKEY MICRO 1,, KEY                                          COMSIDP  165  A
COMSIDP   '?IDSKEY MICCNT '?IDCKEY                                        COMSIDP  166  A
COMSIDP   '?IDSBIT SET '?IDSKEY*CHAR                                      COMSIDP  167  A
COMSIDP   '?IDCHLP MICRO '?IDSKEY+2,, KEY                                 COMSIDP  168  A
COMSIDP  *                                                                COMSIDP  169  A
COMSIDP   '?ID#001 '?IDMKWL                                               COMSIDP  170  A
COMSIDP  *                                                                COMSIDP  171  A
COMSIDP   '?ID#002 IFEQ OI.ERR,2                                          COMSIDP  172  A
COMSIDP   '?IDCS01 MICRO 1,,#=0C "'?IDCHLP" #                             COMSIDP  173  A
COMSIDP   '?ID#002 ELSE                                                   COMSIDP  174  A
COMSIDP   '?IDCS01 MICRO 1,, 0                                            COMSIDP  175  A
COMSIDP   '?ID#002 ENDIF                                                  COMSIDP  176  A
COMSIDP  *                                                                COMSIDP  177  A
COMSIDP    IF -DEF,ST.LAB,1                                               COMSIDP  178  A
COMSIDP   ST.LAB BSS 0                                                    COMSIDP  179  A
COMSIDP  *                                                                COMSIDP  180  A
COMSIDP   LIP.LAB CEQU 0                                                  COMSIDP  181  A
COMSIDP   PTR.LAB CEQU 0                                                  COMSIDP  182  A
COMSIDP   UDO.LAB CEQU 0                                                  COMSIDP  183  A
COMSIDP  *                                                                COMSIDP  184  A
COMSIDP   + VFD KW.PNRL/PTR.LAB,KW.PADL/=YST=LAB,KW.XATL/XA,KW.LENL/'?IDSBIT,____  COMSIDP  185  A
COMSIDP  ,KW.KEYL/'?ID#001                                                COMSIDP  186  A
COMSIDP   + VFD KW.ATTRL/0,KW.LADL/LIP.LAB,KW.UDOL/UDO.LAB,KW.HADL/"'?IDCS01"  COMSIDP  187  A
COMSIDP  *                                                                COMSIDP  188  A
COMSIDP   KEYW     ENDM                                                   COMSIDP  189  A
COMSIDP   LXQ      SPACE  4,8                                             COMSIDP  190  A
COMSIDP  **        LXQ - REDEFINE THE LEFT SHIFT INSTRUCTION.             COMSIDP  191  A
COMSIDP  *                                                                COMSIDP  192  A
COMSIDP  *         THIS OPDEF REDEFINES THE LEFT SHIFT INSTRUCTION TO SUPPRESS  COMSIDP  193  A
COMSIDP  *         CODE GENERATION WHEN THE SHIFT COUNT IS 0, +60D OR -60D.  COMSIDP  194  A
COMSIDP  *         THE INSTRUCTION IS OTHERWISE UNCHANGED.                COMSIDP  195  A
COMSIDP  *                                                                COMSIDP  196  A
COMSIDP  *         LXI       JK                                           COMSIDP  197  A
COMSIDP  *                                                                COMSIDP  198  A
COMSIDP  *         ENTRY  *XI* = X-REG TO BE SHIFTED                      COMSIDP  199  A
COMSIDP  *                *JK* = SHIFT COUNT EXPRESSION                   COMSIDP  200  A
COMSIDP  *                                                                COMSIDP  201  A
COMSIDP  *         USES   XI                                              COMSIDP  202  A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMSIDP

```
COMSIDP                                                                  COMSIDP      203     A
COMSIDP                                                                  COMSIDP      204     A
COMSIDP            PURGDEF ^XQ                                           COMSIDP      205     A
COMSIDP            PURGDEF   LXQ                                         COMSIDP      206     A
COMSIDP   ^XQ      CPOP   0,200B,100B                                    COMSIDP      207     A
COMSIDP                                                                  COMSIDP      208     A
COMSIDP   LXQ      OPDEF     I,JK                                        COMSIDP      209     A
COMSIDP    IFNE JK,0,2                                                   COMSIDP      210     A
COMSIDP    IFNE JK_&60D,0,1                                              COMSIDP      211     A
COMSIDP    ^X.I JK                                                       COMSIDP      212     A
COMSIDP    ENDM                                                          COMSIDP      213     A
COMSIDP   PASOP    SPACE  4,10                                           COMSIDP      214     A
COMSIDP  **        PASOP - DEFINE *POT* TABLE ENTRY.                     COMSIDP      215     A
COMSIDP  *                                                               COMSIDP      216     A
COMSIDP  *                                                               COMSIDP      217     A
COMSIDP  *         THIS MACRO DEFINES A *POT* (PARSING OPERATOR/OPERAND TABLE)  COMSIDP  218     A
COMSIDP  *         ENTRY.  SEE *POT* IN *COMCIDP*.                       COMSIDP      219     A
COMSIDP  *                                                               COMSIDP      220     A
COMSIDP  * TOT     PASOP  POP=A,PIP=B,UPIP=C,PRI=D                       COMSIDP      221     A
COMSIDP  *                                                               COMSIDP      222     A
COMSIDP  *         ENTRY  TOT  = TOKEN TYPE, **WITHOUT** 'O.' PREFIX.    COMSIDP      223     A
COMSIDP  *                POP  = POLISH OUT PROCESSOR ADDR.              COMSIDP      224     A
COMSIDP  *                PIP  = POLISH IN PROCESSOR ADDR.               COMSIDP      225     A
COMSIDP  *                UPIP = POLISH IN PROCESSOR ADDR IF UNARY OPERATION.  COMSIDP  226     A
COMSIDP  *                PRI  = PARSING PRIORITY.                       COMSIDP      227     A
COMSIDP  *                                                               COMSIDP      228     A
COMSIDP  *         EXIT   NONE                                           COMSIDP      229     A
COMSIDP  *                                                               COMSIDP      230     A
COMSIDP  *         USES   NONE                                           COMSIDP      231     A
COMSIDP  *                                                               COMSIDP      232     A
COMSIDP  *         CALLS  NONE                                           COMSIDP      233     A
COMSIDP                                                                  COMSIDP      234     A
COMSIDP                                                                  COMSIDP      235     A
COMSIDP            PURGMAC PASOP                                         COMSIDP      236     A
COMSIDP                                                                  COMSIDP      237     A
COMSIDP            MACROE PASOP,TOT,POP,PIP,UPIP,PRI                     COMSIDP      238     A
COMSIDP  *                                                               COMSIDP      239     A
COMSIDP  *         CHECK FOR *TABLE OUT-OF-ORDER*.                       COMSIDP      240     A
COMSIDP  *                                                               COMSIDP      241     A
COMSIDP    IFLE O.TOT,'?IDSTOT,1                                         COMSIDP      242     A
COMSIDP   P ERR *POT* OUT-OF-ORDER OR DUPLICATE ENTRY                    COMSIDP      243     A
COMSIDP  *                                                               COMSIDP      244     A
COMSIDP    IFNE O.TOT,'?IDSTOT+1,1                                       COMSIDP      245     A
COMSIDP   P ERR *POT* ENTRY MISSING.                                     COMSIDP      246     A
COMSIDP  *                                                               COMSIDP      247     A
COMSIDP   '?IDSTOT SET O.TOT                                             COMSIDP      248     A
COMSIDP  *                                                               COMSIDP      249     A
COMSIDP  *         SET UP DEFAULTS.                                      COMSIDP      250     A
COMSIDP  *                                                               COMSIDP      251     A
COMSIDP   '?IDCS01 MICRO 1,, POP                                         COMSIDP      252     A
COMSIDP    IFC EQ, POP  ,1                                               COMSIDP      253     A
COMSIDP   '?IDCS01 MICRO 1,, E.POP                                       COMSIDP      254     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMSIDP

```
COMSIDP  *                                                                COMSIDP    255    A
COMSIDP   '?IDCS02 MICRO 1,, PIP                                          COMSIDP    256    A
COMSIDP    IFC EQ, PIP  ,1                                                COMSIDP    257    A
COMSIDP   '?IDCS02 MICRO 1,, SE.NPP                                       COMSIDP    258    A
COMSIDP  *                                                                COMSIDP    259    A
COMSIDP   '?IDCS03 MICRO 1,, UPIP                                         COMSIDP    260    A
COMSIDP    IFC EQ, UPIP  ,1                                               COMSIDP    261    A
COMSIDP   '?IDCS03 MICRO 1,, E.UNOP                                       COMSIDP    262    A
COMSIDP  *                                                                COMSIDP    263    A
COMSIDP  *         GENERATE *POT* ENTRY.                                  COMSIDP    264    A
COMSIDP  *                                                                COMSIDP    265    A
COMSIDP   + VFD PS.PRIL/PRI,PS.UPIPL/"'?IDCS03",PS.PIPL/"'?IDCS02",PS.POPL/______  COMSIDP    266    A
COMSIDP  ,___"'?IDCS01"                                                   COMSIDP    267    A
COMSIDP  *                                                                COMSIDP    268    A
COMSIDP   PASOP    ENDM                                                   COMSIDP    269    A
COMSIDP   PRBDO    SPACE  4,8                                             COMSIDP    270    A
COMSIDP  **        PRBDO - PRINT CODED LINE ON BATCH DEBUG OUTPUT FILE.   COMSIDP    271    A
COMSIDP  *                                                                COMSIDP    272    A
COMSIDP  *                                                                COMSIDP    273    A
COMSIDP  *                IF THE USER DID NOT DEFINE A BATCH DEBUG OUTPUT FILE,  COMSIDP    274    A
COMSIDP  *         (F.BDO -DEF), THEN *IDP* WILL INVENT ONE AND ACCESS IT VIA  COMSIDP    275    A
COMSIDP  *         *PRBDO* MACRO. IF *F.BDO* IS DEFINED, THEN THE USER SUPPLIES  COMSIDP    276    A
COMSIDP  *         HER OWN *PRBDO*.                                       COMSIDP    277    A
COMSIDP  *                                                                COMSIDP    278    A
COMSIDP  *         PRBDO  FWA,LEN                                         COMSIDP    279    A
COMSIDP  *                                                                COMSIDP    280    A
COMSIDP  *         ENTRY  FWA = FWA OF LINE (-C- FORMAT)                  COMSIDP    281    A
COMSIDP  *                LEN = LENGTH OF LINE (IN WORDS)                 COMSIDP    282    A
COMSIDP  *                    = .ZR. IF LENGTH TO BE COMPUTED             COMSIDP    283    A
COMSIDP  *                                                                COMSIDP    284    A
COMSIDP  *         EXIT   NONE                                            COMSIDP    285    A
COMSIDP  *                                                                COMSIDP    286    A
COMSIDP  *         USES   ALL BUT A0,X0,A5,X5   (INCLUDES ALL CALLS)      COMSIDP    287    A
COMSIDP  *                                                                COMSIDP    288    A
COMSIDP  *         CALLS  WRITEC                                          COMSIDP    289    A
COMSIDP                                                                   COMSIDP    290    A
COMSIDP                                                                   COMSIDP    291    A
COMSIDP   #BDO     IF     -MAC,PRBDO                                      COMSIDP    292    A
COMSIDP                                                                   COMSIDP    293    A
COMSIDP            PURGMAC PRBDO                                          COMSIDP    294    A
COMSIDP                                                                   COMSIDP    295    A
COMSIDP   PRBDO    MACRO  FWA,LEN                                         COMSIDP    296    A
COMSIDP            WRITEC =XF.BDO,(FWA),(LEN)                             COMSIDP    297    A
COMSIDP   PRBDO    ENDM                                                   COMSIDP    298    A
COMSIDP   #BDO     ENDIF                                                  COMSIDP    299    A
COMSIDP   PRIDP    SPACE  4,8                                             COMSIDP    300    A
COMSIDP  **        PRIDP - PRINT CODED LINE ON OUTPUT FILE.               COMSIDP    301    A
COMSIDP  *                                                                COMSIDP    302    A
COMSIDP  *                                                                COMSIDP    303    A
COMSIDP  *         PRIDP  FWA,LEN,NRB                                     COMSIDP    304    A
COMSIDP  *                                                                COMSIDP    305    A
COMSIDP  *         ENTRY  FWA = FWA OF LINE (-C- FORMAT)                  COMSIDP    306    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMSIDP

```
COMSIDP  *                LEN = LENGTH OF LINE (IN WORDS)                          COMSIDP    307    A
COMSIDP  *                    = .ZR. IF LENGTH TO BE COMPUTED                      COMSIDP    308    A
COMSIDP  *                NRB = NR OF BLANK LINES TO OUTPUT PRECEDING THE LINE     COMSIDP    309    A
COMSIDP  *                                                                         COMSIDP    310    A
COMSIDP  *         EXIT   NONE                                                     COMSIDP    311    A
COMSIDP  *                                                                         COMSIDP    312    A
COMSIDP  *         USES   ALL BUT B4,A0,X0,A5,X5 (INCLUDES ALL CALLS)              COMSIDP    313    A
COMSIDP  *                                                                         COMSIDP    314    A
COMSIDP  *         CALLS  ROL                                                      COMSIDP    315    A
COMSIDP                                                                            COMSIDP    316    A
COMSIDP                                                                            COMSIDP    317    A
COMSIDP            PURGMAC PRIDP                                                   COMSIDP    318    A
COMSIDP                                                                            COMSIDP    319    A
COMSIDP   PRIDP    MACRO  FWA,LEN,NRB                                              COMSIDP    320    A
COMSIDP            =X6    FWA                                                      COMSIDP    321    A
COMSIDP            =X7    LEN                                                      COMSIDP    322    A
COMSIDP            =X4    NRB                                                      COMSIDP    323    A
COMSIDP            RJ     =XROL=                                                   COMSIDP    324    A
COMSIDP   PRIDP    ENDM                                                            COMSIDP    325    A
COMSIDP   REQUEST  SPACE  4,10                                                     COMSIDP    326    A
COMSIDP  **        REQUEST - REQUEST OPERATOR ASSIGNMENT OF EQUIPMENT.             COMSIDP    327    A
COMSIDP  *                                                                         COMSIDP    328    A
COMSIDP  *                                                                         COMSIDP    329    A
COMSIDP  *         THIS MACRO IS FOR *KRONOS/NOS* **ONLY**.  IT EXISTS BECAUSE     COMSIDP    330    A
COMSIDP  *         OF A CONFLICT BETWEEN THE *REQUEST* MACRO DEFINITION IN         COMSIDP    331    A
COMSIDP  *         *CPUTEXT* (WHICH IS FOR *SCOPE*) AND ANY REFERENCES TO THIS     COMSIDP    332    A
COMSIDP  *         MACRO ON *KRONOS*.  I.E. THEY ARE INCOMPATIBLE.                 COMSIDP    333    A
COMSIDP  *                                                                         COMSIDP    334    A
COMSIDP  *         COPIED FROM *OPL/CPCOM*.                                        COMSIDP    335    A
COMSIDP  *                                                                         COMSIDP    336    A
COMSIDP  *         REQUEST FILE,UNIT                                               COMSIDP    337    A
COMSIDP  *                                                                         COMSIDP    338    A
COMSIDP  *         ENTRY  *UNIT* IF USED ASSIGNS THE EQUIPMENT DESIGNATED          COMSIDP    339    A
COMSIDP  *                IN THE *DTY* POSITION IN (FILE+1).                       COMSIDP    340    A
COMSIDP  *                                                                         COMSIDP    341    A
COMSIDP  *         CALLS  LFM=                                                     COMSIDP    342    A
COMSIDP                                                                            COMSIDP    343    A
COMSIDP                                                                            COMSIDP    344    A
COMSIDP   #OS1     IFEQ   .OS,1       IF *KRONOS/NOS*                              COMSIDP    345    A
COMSIDP            PURGMAC REQUEST                                                 COMSIDP    346    A
COMSIDP                                                                            COMSIDP    347    A
COMSIDP   REQUEST  MACRO  F,U                                                      COMSIDP    348    A
COMSIDP    R= X2,F                                                                 COMSIDP    349    A
COMSIDP   '?ID#001 IFC EQ,$U$$                                                     COMSIDP    350    A
COMSIDP    SX7 14B                                                                 COMSIDP    351    A
COMSIDP   '?ID#001 ELSE                                                            COMSIDP    352    A
COMSIDP    SX7 15B                                                                 COMSIDP    353    A
COMSIDP   '?ID#001 ENDIF                                                           COMSIDP    354    A
COMSIDP    RJ =XLFM=                                                               COMSIDP    355    A
COMSIDP   REQUEST ENDM                                                             COMSIDP    356    A
COMSIDP   #OS1     ENDIF                                                           COMSIDP    357    A
COMSIDP   SCTOT    SPACE  4,10                                                     COMSIDP    358    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMSIDP

```
COMSIDP  **        SCTOT - DEFINE *STMT CONTROL* TABLE ENTRY.                    COMSIDP     359     A
COMSIDP  *                                                                       COMSIDP     360     A
COMSIDP  *                                                                       COMSIDP     361     A
COMSIDP  * TOT     SCTOT  PAD                                                    COMSIDP     362     A
COMSIDP  *                                                                       COMSIDP     363     A
COMSIDP  *         ENTRY  TOT = TOKEN TYPE (WITHOUT *O.* PREFIX*).               COMSIDP     364     A
COMSIDP  *                PAD = PROCESSOR ADDR FOR THIS *TOT*.                   COMSIDP     365     A
COMSIDP  *                                                                       COMSIDP     366     A
COMSIDP  *         EXIT   NONE                                                   COMSIDP     367     A
COMSIDP  *                                                                       COMSIDP     368     A
COMSIDP  *         USES   NONE                                                   COMSIDP     369     A
COMSIDP  *                                                                       COMSIDP     370     A
COMSIDP  *         CALLS  NONE                                                   COMSIDP     371     A
COMSIDP                                                                          COMSIDP     372     A
COMSIDP                                                                          COMSIDP     373     A
COMSIDP            PURGMAC SCTOT                                                 COMSIDP     374     A
COMSIDP                                                                          COMSIDP     375     A
COMSIDP            MACRO  SCTOT,TOT,PAD                                          COMSIDP     376     A
COMSIDP  *                                                                       COMSIDP     377     A
COMSIDP   '?IDS001 SET    SC.TOTP-SC.PADP-SC.PADL                                COMSIDP     378     A
COMSIDP   + VFD SC.TOTL/O.TOT,'?IDS001/0,SC.PADL/PAD                             COMSIDP     379     A
COMSIDP  *                                                                       COMSIDP     380     A
COMSIDP   SCTOT    ENDM                                                          COMSIDP     381     A
COMSIDP   SUBKEY   SPACE  4,10                                                   COMSIDP     382     A
COMSIDP  **        SUBKEY - GENERATE SUB-KEYWORD TABLE ENTRY.                    COMSIDP     383     A
COMSIDP  *                                                                       COMSIDP     384     A
COMSIDP  *                                                                       COMSIDP     385     A
COMSIDP  *         MACRO USED TO GENERATE AN IDP SUB-KEYWORD ENTRY.              COMSIDP     386     A
COMSIDP  *                                                                       COMSIDP     387     A
COMSIDP  *         SUBKEY KEY,VAL                                                COMSIDP     388     A
COMSIDP  *                                                                       COMSIDP     389     A
COMSIDP  *         ENTRY  KEY  = SUB-KEYWORD.                                    COMSIDP     390     A
COMSIDP  *                VAL  = VALUE TO BE PLACED IN LOWER 18 BITS OF *SUBKEY* COMSIDP     391     A
COMSIDP  *                       ENTRY.                                          COMSIDP     392     A
COMSIDP  *                                                                       COMSIDP     393     A
COMSIDP  *         EXIT   NONE                                                   COMSIDP     394     A
COMSIDP  *                                                                       COMSIDP     395     A
COMSIDP  *         USES   NONE                                                   COMSIDP     396     A
COMSIDP  *                                                                       COMSIDP     397     A
COMSIDP  *         CALLS  NONE                                                   COMSIDP     398     A
COMSIDP                                                                          COMSIDP     399     A
COMSIDP                                                                          COMSIDP     400     A
COMSIDP            PURGMAC  SUBKEY                                               COMSIDP     401     A
COMSIDP                                                                          COMSIDP     402     A
COMSIDP   SUBKEY   MACRO  KEY,VAL                                                COMSIDP     403     A
COMSIDP            VFD    42/0L_KEY,18/VAL                                       COMSIDP     404     A
COMSIDP   SUBKEY   ENDM                                                          COMSIDP     405     A
COMSIDP   SYSERR   SPACE  4,10                                                   COMSIDP     406     A
COMSIDP  **        SYSERR - SYSTEM ERROR.                                        COMSIDP     407     A
COMSIDP  *                                                                       COMSIDP     408     A
COMSIDP  *                                                                       COMSIDP     409     A
COMSIDP  * LOC     SYSERR MSG,JMP                                                COMSIDP     410     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMSIDP

```
COMSIDP  *                                                                  COMSIDP     411    A
COMSIDP  *         ENTRY  LOC = LOCATION FIELD TO JUMP TO FOR THIS SYS ERR. COMSIDP     412    A
COMSIDP  *                MSG = ERROR MSG TEXT.                            COMSIDP     413    A
COMSIDP  *                JMP = ADDR TO JUMP TO FOR THIS SYS ERR,          COMSIDP     414    A
COMSIDP  *                      (DEFAULT = *IDP=SER*).                     COMSIDP     415    A
COMSIDP  *                                                                  COMSIDP     416    A
COMSIDP  *         EXIT   NONE                                              COMSIDP     417    A
COMSIDP  *                                                                  COMSIDP     418    A
COMSIDP  *         USES   NONE                                              COMSIDP     419    A
COMSIDP  *                                                                  COMSIDP     420    A
COMSIDP  *         CALLS  NONE                                              COMSIDP     421    A
COMSIDP                                                                     COMSIDP     422    A
COMSIDP                                                                     COMSIDP     423    A
COMSIDP            PURGMAC SYSERR                                           COMSIDP     424    A
COMSIDP                                                                     COMSIDP     425    A
COMSIDP            MACRO  SYSERR,LOC,MSG,JMP                                COMSIDP     426    A
COMSIDP  *                                                                  COMSIDP     427    A
COMSIDP   LOC BSS 0                                                         COMSIDP     428    A
COMSIDP  *                                                                  COMSIDP     429    A
COMSIDP    IF -DEF,'?IDSSER,1                                               COMSIDP     430    A
COMSIDP   '?IDSSER SET 0                                                    COMSIDP     431    A
COMSIDP  *                                                                  COMSIDP     432    A
COMSIDP   '?IDSSER SET '?IDSSER+1                                           COMSIDP     433    A
COMSIDP  *                                                                  COMSIDP     434    A
COMSIDP   '?IDCS01 MICRO 1,,# MSG_#                                         COMSIDP     435    A
COMSIDP   '?IDS001 MICCNT '?IDCS01                                          COMSIDP     436    A
COMSIDP    IFGT '?IDS001,70D,1                                              COMSIDP     437    A
COMSIDP   P ERR ERR MESSAGE TOO LONG.                                       COMSIDP     438    A
COMSIDP  *                                                                  COMSIDP     439    A
COMSIDP    SX6 =C# MSG_#                                                    COMSIDP     440    A
COMSIDP    EQ JMP IDP=SER                                                   COMSIDP     441    A
COMSIDP  *                                                                  COMSIDP     442    A
COMSIDP   SYSERR   ENDM                                                     COMSIDP     443    A
COMSIDP   '?IDMKWL SPACE  4,10                                              COMSIDP     444    A
COMSIDP  **        '?IDMKWL - GENERATE KEYWORD LITERAL.                     COMSIDP     445    A
COMSIDP  *                                                                  COMSIDP     446    A
COMSIDP  *                                                                  COMSIDP     447    A
COMSIDP  * LAB     '?IDMKWL                                                 COMSIDP     448    A
COMSIDP  *                                                                  COMSIDP     449    A
COMSIDP  *         ENTRY  '?IDCKEY = MICRO CONTAINING KEYWORD.              COMSIDP     450    A
COMSIDP  *                '?IDSKEY = LEN OF '?IDCKEY (IN CHARS, I.E. *MICCNT*). COMSIDP 451    A
COMSIDP  *                LAB      = LABEL FIELD FOR THIS KEYWORD LITERAL.  COMSIDP     452    A
COMSIDP  *                                                                  COMSIDP     453    A
COMSIDP  *         EXIT   NONE                                              COMSIDP     454    A
COMSIDP  *                                                                  COMSIDP     455    A
COMSIDP  *         USES   NONE                                              COMSIDP     456    A
COMSIDP  *                                                                  COMSIDP     457    A
COMSIDP  *         CALLS  NONE                                              COMSIDP     458    A
COMSIDP                                                                     COMSIDP     459    A
COMSIDP                                                                     COMSIDP     460    A
COMSIDP            PURGMAC '?IDMKWL                                         COMSIDP     461    A
COMSIDP                                                                     COMSIDP     462    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMSIDP

```
COMSIDP            MACRO  '?IDMKWL,LAB                                    COMSIDP     463    A
COMSIDP    LOCAL '?ID#001                                                 COMSIDP     464    A
COMSIDP  *                                                                COMSIDP     465    A
COMSIDP   '?IDCS02 MICRO 1,, 0L                                           COMSIDP     466    A
COMSIDP   '?IDS001 SET '?IDSKEY/7+1                                       COMSIDP     467    A
COMSIDP   '?IDS002 SET 1                                                  COMSIDP     468    A
COMSIDP  *                                                                COMSIDP     469    A
COMSIDP   '?ID#001 DUP '?IDS001                                           COMSIDP     470    A
COMSIDP   '?IDCS01 MICRO '?IDS002,7, "'?IDCKEY"                           COMSIDP     471    A
COMSIDP   '?IDCS02 MICRO 1,, "'?IDCS02",0L"'?IDCS01"                      COMSIDP     472    A
COMSIDP   '?IDS002 SET '?IDS002+7                                         COMSIDP     473    A
COMSIDP  *                                                                COMSIDP     474    A
COMSIDP   '?ID#001 ENDD                                                   COMSIDP     475    A
COMSIDP  *                                                                COMSIDP     476    A
COMSIDP   LAB LIT "'?IDCS02"                                              COMSIDP     477    A
COMSIDP  *                                                                COMSIDP     478    A
COMSIDP   '?IDMKWL ENDM                                                   COMSIDP     479    A
COMSIDP   '?IDMLIT SPACE  4,10                                            COMSIDP     480    A
COMSIDP  **        '?IDMLIT - GENERATE IDP LITERAL.                       COMSIDP     481    A
COMSIDP  *                                                                COMSIDP     482    A
COMSIDP  *                                                                COMSIDP     483    A
COMSIDP  *         THIS MACRO GENERATES A LITERAL TO IDP.S LITERAL BLOCK, COMSIDP     484    A
COMSIDP  *         '?IDRLIT.  *'?IDMLIT*, THEN, SIMULATES THE COMPASS *LIT* COMSIDP     485    A
COMSIDP  *         PSEUDO, WITH THE PRIMARY DIFFERENCE BEING THAT *'?IDMLIT* COMSIDP     486    A
COMSIDP  *         DOES NOT GENERATE TO THE PROGRAM LITERAL BLOCK.        COMSIDP     487    A
COMSIDP  *                                                                COMSIDP     488    A
COMSIDP  *         THIS SHOULD HELP TO MINIMIZE UNFAVOURABLE IDP/HOST     COMSIDP     489    A
COMSIDP  *         INTERACTIONS.                                          COMSIDP     490    A
COMSIDP  *                                                                COMSIDP     491    A
COMSIDP  * LOC     '?IDMLIT LIT                                          COMSIDP     492    A
COMSIDP  *                                                                COMSIDP     493    A
COMSIDP  *         ENTRY  LOC = ADDR/LOCATION FIELD FOR THIS LITERAL.     COMSIDP     494    A
COMSIDP  *                LIT = LITERAL STRING.                           COMSIDP     495    A
COMSIDP  *                                                                COMSIDP     496    A
COMSIDP  *         EXIT   NONE                                            COMSIDP     497    A
COMSIDP  *                                                                COMSIDP     498    A
COMSIDP  *         USES   NONE                                            COMSIDP     499    A
COMSIDP  *                                                                COMSIDP     500    A
COMSIDP  *         CALLS  NONE                                            COMSIDP     501    A
COMSIDP                                                                   COMSIDP     502    A
COMSIDP                                                                   COMSIDP     503    A
COMSIDP            PURGMAC '?IDMLIT                                       COMSIDP     504    A
COMSIDP                                                                   COMSIDP     505    A
COMSIDP            MACRO  '?IDMLIT,LOC,LIT                                COMSIDP     506    A
COMSIDP  *                                                                COMSIDP     507    A
COMSIDP   '?IDCS01 MICRO 1,,#_LIT_#                                       COMSIDP     508    A
COMSIDP    '?IDMSMA 1,4                                                   COMSIDP     509    A
COMSIDP  *                                                                COMSIDP     510    A
COMSIDP   '?ID#001 IFEQ '?IDS001,0                                        COMSIDP     511    A
COMSIDP   '?ID1LEN SET '?ID1LEN+1                                         COMSIDP     512    A
COMSIDP  *                                                                COMSIDP     513    A
COMSIDP    IFGT '?ID1LEN,999D,1                                           COMSIDP     514    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMSIDP

```
COMSIDP   P ERR TOO MANY LITERALS                                      COMSIDP      515     A
COMSIDP  *                                                             COMSIDP      516     A
COMSIDP   '?IDCS02 DECMIC '?ID1LEN,3                                   COMSIDP      517     A
COMSIDP   '?ID1"'?IDCS02" MICRO 1,,#"'?IDCS01"#                        COMSIDP      518     A
COMSIDP  *                                                             COMSIDP      519     A
COMSIDP   '?IDS001 SET '?ID1LEN                                        COMSIDP      520     A
COMSIDP   '?IDRLIT '?IDMRMT (  DATA LIT)                               COMSIDP      521     A
COMSIDP  *                                                             COMSIDP      522     A
COMSIDP   '?ID#001 ENDIF                                               COMSIDP      523     A
COMSIDP  *                                                             COMSIDP      524     A
COMSIDP   LOC = FW.L_I_T+'?IDS001                                      COMSIDP      525     A
COMSIDP  *                                                             COMSIDP      526     A
COMSIDP   '?IDMLIT ENDM                                                COMSIDP      527     A
COMSIDP   '?IDMMEQ SPACE  4,10                                         COMSIDP      528     A
COMSIDP  **        '?IDMMEQ - MICRO EQUIVALENCE.                       COMSIDP      529     A
COMSIDP  *                                                             COMSIDP      530     A
COMSIDP  *                                                             COMSIDP      531     A
COMSIDP  *         THIS MACRO EQUATES TWO MICRO STRINGS.  IT IS ONLY TRULY  COMSIDP  532     A
COMSIDP  *         USEFUL WHEN THE MICRO NAME ON THE RIGHT-SIDE OF THE  COMSIDP      533     A
COMSIDP  *         EQUIVALENCE IS BEING GENERATED VIA MICRO STRINGS.   COMSIDP      534     A
COMSIDP  *                                                             COMSIDP      535     A
COMSIDP  * A       '?IDMMEQ B                                          COMSIDP      536     A
COMSIDP  *                                                             COMSIDP      537     A
COMSIDP  *         ENTRY  A = MICRO NAME TO EQUATE *B* TO.             COMSIDP      538     A
COMSIDP  *                B = OLD MICRO NAME.                          COMSIDP      539     A
COMSIDP  *                                                             COMSIDP      540     A
COMSIDP  *         EXIT   "A" CREATED AND "B" LEFT ALONE.              COMSIDP      541     A
COMSIDP  *                                                             COMSIDP      542     A
COMSIDP  *         CALLS  NONE                                         COMSIDP      543     A
COMSIDP                                                                COMSIDP      544     A
COMSIDP                                                                COMSIDP      545     A
COMSIDP            PURGMAC '?IDMMEQ                                    COMSIDP      546     A
COMSIDP                                                                COMSIDP      547     A
COMSIDP    MACRO '?IDMMEQ,A,B                                          COMSIDP      548     A
COMSIDP   A MICRO 1,, "B"                                              COMSIDP      549     A
COMSIDP   '?IDMMEQ ENDM                                                COMSIDP      550     A
COMSIDP   '?IDMRMT SPACE  4,10                                         COMSIDP      551     A
COMSIDP  **        '?IDMRMT - FORCE MICRO EVALUATION FOR *RMT* BLOCKS. COMSIDP      552     A
COMSIDP  *                                                             COMSIDP      553     A
COMSIDP  *                                                             COMSIDP      554     A
COMSIDP  *         THIS MACRO IS USED SOLEY FOR THE PURPOSE OF FORCING COMSIDP      555     A
COMSIDP  *         MICRO EVALUATION IN *RMT* BLOCKS.                   COMSIDP      556     A
COMSIDP  *                                                             COMSIDP      557     A
COMSIDP  * BLK     '?IDMRMT S                                          COMSIDP      558     A
COMSIDP  *                                                             COMSIDP      559     A
COMSIDP  *         ENTRY  BLK = REMOTE BLOCK NAME                      COMSIDP      560     A
COMSIDP  *                S   = STMT TO BE ENTERED INTO REMOTE BLOCK *BLK*.  COMSIDP  561     A
COMSIDP  *                                                             COMSIDP      562     A
COMSIDP  *         EXIT   NONE                                         COMSIDP      563     A
COMSIDP  *                                                             COMSIDP      564     A
COMSIDP  *         USES   NONE                                         COMSIDP      565     A
COMSIDP  *                                                             COMSIDP      566     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMSIDP

```
COMSIDP  *         CALLS  NONE                                                  COMSIDP      567     A
COMSIDP                                                                         COMSIDP      568     A
COMSIDP                                                                         COMSIDP      569     A
COMSIDP            PURGMAC '?IDMRMT                                             COMSIDP      570     A
COMSIDP                                                                         COMSIDP      571     A
COMSIDP            MACRO  '?IDMRMT,BLK,S                                        COMSIDP      572     A
COMSIDP   BLK RMT                                                               COMSIDP      573     A
COMSIDP   S                                                                     COMSIDP      574     A
COMSIDP   BLK RMT                                                               COMSIDP      575     A
COMSIDP   '?IDMRMT ENDM                                                         COMSIDP      576     A
COMSIDP   '?IDMSMA SPACE  4,10                                                  COMSIDP      577     A
COMSIDP  **        '?IDMSMA - SCAN MICRO ARRAY.                                 COMSIDP      578     A
COMSIDP  *                                                                      COMSIDP      579     A
COMSIDP  *                                                                      COMSIDP      580     A
COMSIDP  *         '?IDMSMA N,FC,NC                                             COMSIDP      581     A
COMSIDP  *                                                                      COMSIDP      582     A
COMSIDP  *         ENTRY  N  - MICRO ARRAY NR.                                  COMSIDP      583     A
COMSIDP  *                FC = 1ST CHAR POSITION IN EACH ARRAY ELEMENT          COMSIDP      584     A
COMSIDP  *                     TO SEARCH AT.                                    COMSIDP      585     A
COMSIDP  *                NC = NR OF CHARS (BEGINNING AT *FC*) TO COMPARE,      COMSIDP      586     A
COMSIDP  *                   = 0 IF REST OF MICRO ARRAY ENTRY (AS IN *MICRO*    COMSIDP      587     A
COMSIDP  *                     PSEUDO).                                         COMSIDP      588     A
COMSIDP  *                                                                      COMSIDP      589     A
COMSIDP  *                '?IDCS01 = ELEMENT TO SEARCH FOR.                     COMSIDP      590     A
COMSIDP  *                '?IDNLEN = NR OF ENTRIES IN THIS MICRO ARRAY          COMSIDP      591     A
COMSIDP  *                           (WHERE *N* IS MICRO ARRAY NR).             COMSIDP      592     A
COMSIDP  *                                                                      COMSIDP      593     A
COMSIDP  *         EXIT   '?IDCS01 = UNCHANGED.                                 COMSIDP      594     A
COMSIDP  *                '?IDS001 = MICRO ARRAY ELEMENT NR IF A FIND, ELSE     COMSIDP      595     A
COMSIDP  *                         = .ZR. IF NO FIND.                           COMSIDP      596     A
COMSIDP  *                '?IDCS02 = MICRO CONTAINING FOUND ENTRY IN '?IDNXXX.  COMSIDP      597     A
COMSIDP  *                '?IDCS03 = MICRO CONTAINING MICRO NAME OF FOUND       COMSIDP      598     A
COMSIDP  *                           ENTRY.                                     COMSIDP      599     A
COMSIDP  *                           I.E. "'?IDCS03" MICRO 1,, "'?IDCS02"       COMSIDP      600     A
COMSIDP  *                                                                      COMSIDP      601     A
COMSIDP  *         CALLS  '?IDMMEQ                                              COMSIDP      602     A
COMSIDP                                                                         COMSIDP      603     A
COMSIDP                                                                         COMSIDP      604     A
COMSIDP            PURGMAC '?IDMSMA                                             COMSIDP      605     A
COMSIDP                                                                         COMSIDP      606     A
COMSIDP   '?IDMSMA MACRO A,B,C                                                  COMSIDP      607     A
COMSIDP    LOCAL '?ID#001,'?ID#002,'?ID#003                                     COMSIDP      608     A
COMSIDP  *                                                                      COMSIDP      609     A
COMSIDP   '?IDS001 SET '?ID_A_LEN                                               COMSIDP      610     A
COMSIDP   '?ID#002 IFGT '?IDS001,0                                              COMSIDP      611     A
COMSIDP  *                                                                      COMSIDP      612     A
COMSIDP   '?ID#001 DUP '?ID_A_LEN                                               COMSIDP      613     A
COMSIDP  *                                                                      COMSIDP      614     A
COMSIDP   '?IDCS03 DECMIC '?IDS001,3                                            COMSIDP      615     A
COMSIDP   '?IDCS03 MICRO 1,, '?ID_A_"'?IDCS03"                                  COMSIDP      616     A
COMSIDP   '?IDCS02 '?IDMMEQ "'?IDCS03"                                          COMSIDP      617     A
COMSIDP   '?IDCS04 MICRO B,C, "'?IDCS02"                                        COMSIDP      618     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMSIDP

```
COMSIDP   '?ID#003  IFC EQ, "'?IDCS01" "'?IDCS04"                    COMSIDP    619    A
COMSIDP    STOPDUP                                                   COMSIDP    620    A
COMSIDP  *                                                           COMSIDP    621    A
COMSIDP   '?ID#003 ELSE                                              COMSIDP    622    A
COMSIDP   '?IDS001 SET '?IDS001-1                                    COMSIDP    623    A
COMSIDP   '?ID#003 ENDIF                                             COMSIDP    624    A
COMSIDP   '?ID#001 ENDD                                              COMSIDP    625    A
COMSIDP   '?ID#002 ENDIF                                             COMSIDP    626    A
COMSIDP  *                                                           COMSIDP    627    A
COMSIDP   '?IDMSMA ENDM                                              COMSIDP    628    A
COMSIDP   MICROS   SPACE  4,8                                        COMSIDP    629    A
COMSIDP  **        DEFINE DEFAULT MICROS.                            COMSIDP    630    A
COMSIDP                                                              COMSIDP    631    A
COMSIDP            IF     -MIC,EOS,1                                 COMSIDP    632    A
COMSIDP   EOS      MICRO  1,,/;/                                     COMSIDP    633    A
COMSIDP                                                              COMSIDP    634    A
COMSIDP                                                              COMSIDP    635    A
COMSIDP            IF     -MIC,IDPVER,1                              COMSIDP    636    A
COMSIDP   IDPVER   MICRO  1,,/1.1/                                   COMSIDP    637    A
COMSIDP                                                              COMSIDP    638    A
COMSIDP            IF     -MIC,PROMPT,1                              COMSIDP    639    A
COMSIDP   PROMPT   MICRO  1,,/>>/                                    COMSIDP    640    A
COMSIDP   OI.      SPACE  4,10                                       COMSIDP    641    A
COMSIDP  **        OI. - IDP OPTIONS.                                COMSIDP    642    A
COMSIDP  *                                                           COMSIDP    643    A
COMSIDP  *                                                           COMSIDP    644    A
COMSIDP  *         SYMBOLS OF THE FORM *OI.XXX*, WHERE *OI.* IS A COMMON PREFIX  COMSIDP    645    A
COMSIDP  *         AND *XXX* IS A UNIQUE OPTION NAME, DEFINE MISCELLANEOUS IDP   COMSIDP    646    A
COMSIDP  *         OPTIONS.                                          COMSIDP    647    A
COMSIDP  *                                                           COMSIDP    648    A
COMSIDP  *         MOST OF THESE OPTIONS ARE RELATED TO THE CONFIGURING OF IDP   COMSIDP    649    A
COMSIDP  *         AND ITS SUPPORTING COMDECKS.  TO SET ONE OR MORE OF THESE     COMSIDP    650    A
COMSIDP  *         OPTIONS, THE IDP INSTALLER NEED ONLY TO DEFINE THE APPROPRIATE COMSIDP    651    A
COMSIDP  *         DESIRED *OI.XXX* SYMBOL.  IDP DOES THE REST.      COMSIDP    652    A
COMSIDP  *                                                           COMSIDP    653    A
COMSIDP  *         OPTIONS, *XXX*, ARE --                            COMSIDP    654    A
COMSIDP  *                                                           COMSIDP    655    A
COMSIDP  *         ERR  = SELECT/DESELECT ERROR MESSAGES.            COMSIDP    656    A
COMSIDP  *              = 0 MEANS *DESELECT ERROR MESSAGES*.  THAT IS, IF AN COMSIDP    657    A
COMSIDP  *                  ERROR IS DETECTED BY IDP, ALL ONE GETS IS THE ERROR COMSIDP    658    A
COMSIDP  *                  PROMPT.                                  COMSIDP    659    A
COMSIDP  *                                                           COMSIDP    660    A
COMSIDP  *              = 1 MEANS *SELECT SHORT MESSAGES*.  THAT IS, IF AN ERROR COMSIDP    661    A
COMSIDP  *                  IS DETECTED BY IDP, ONE GETS THE ERROR MSG NUMBER. COMSIDP    662    A
COMSIDP  *                                                           COMSIDP    663    A
COMSIDP  *              = 2 (DEFAULT) MEANS *SELECT FULL ERROR MESSAGES*.  IF AN COMSIDP    664    A
COMSIDP  *                  ERROR IS DETECTED IN THIS MODE, IDP ISSUES THE FULL COMSIDP    665    A
COMSIDP  *                  ERROR MSG.                               COMSIDP    666    A
COMSIDP  *                                                           COMSIDP    667    A
COMSIDP  *                THE MAIN DIFFERENCE BETWEEN THESE DIFFERENT *ERR* COMSIDP    668    A
COMSIDP  *                OPTIONS IS ONE OF SPACE (I.E. IDP REQUIRES MORE COMSIDP    669    A
COMSIDP  *                CM STORAGE TO HOLD THE ERROR MESSAGES).  IT IS ADVISED COMSIDP    670    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMSIDP

```
COMSIDP  *                THAT ONE USE/INSTALL IDP WITH FULL ERROR MSGS...AFTER          COMSIDP     671    A
COMSIDP  *                ALL, ONE IS USING IDP TO MAKE LIFE EASIER FOR ONESELF.         COMSIDP     672    A
COMSIDP                                                                                  COMSIDP     673    A
COMSIDP                                                                                  COMSIDP     674    A
COMSIDP   OI.ERR   CEQU   2           INDICATE FULL ERR MSGS                             COMSIDP     675    A
COMSIDP   DEFS     SPACE  4,8                                                            COMSIDP     676    A
COMSIDP  **        MISCELLANEOUS SYMBOL DEFINITIONS.                                     COMSIDP     677    A
COMSIDP                                                                                  COMSIDP     678    A
COMSIDP                                                                                  COMSIDP     679    A
COMSIDP   CHAR     =      6           LENGTH OF CYBER CHARACTER                          COMSIDP     680    A
COMSIDP                                                                                  COMSIDP     681    A
COMSIDP   DXB1$    =      1           SET TO *ALLOW /COMCDXB/ TO CONVERT DIGITS          COMSIDP     682    A
COMSIDP  *                            WITH AN 8 OR 9 BUT NO EXPLICIT *B* OR *D*          COMSIDP     683    A
COMSIDP  *                            SUFFIX TO DEFAULT BASE DECIMAL*                    COMSIDP     684    A
COMSIDP                                                                                  COMSIDP     685    A
COMSIDP   L.BSL    =      4           NR OF CM WORDS DUMPED ON LINE (BATCH)              COMSIDP     686    A
COMSIDP   L.ISL    =      2           NR OF CM WORDS DUMPED ON LINE (INTERACTIVE)        COMSIDP     687    A
COMSIDP                                                                                  COMSIDP     688    A
COMSIDP   L.PRB    =      0           NR OF PRECEDING BLANKS TO OUTPUT                   COMSIDP     689    A
COMSIDP  *                              FOR SPACING PURPOSES                             COMSIDP     690    A
COMSIDP   SVRDEFS  SPACE  4,10                                                           COMSIDP     691    A
COMSIDP  **        *COMCSVR* DEFINITIONS.                                                COMSIDP     692    A
COMSIDP                                                                                  COMSIDP     693    A
COMSIDP                                                                                  COMSIDP     694    A
COMSIDP   #SVR     =      1           SELECT *IDP USES ITS OWN REG SAVE AREA*            COMSIDP     695    A
COMSIDP   BAX      MICRO  1,,/BAX/                                                       COMSIDP     696    A
COMSIDP   DEFS     SPACE  4,8                                                            COMSIDP     697    A
COMSIDP  **        SNAP PARAMETER LIST SYMBOL DEFINITIONS.                               COMSIDP     698    A
COMSIDP  *           THESE SYMBOLS DEFINE THE ORDINALS INTO THE SNAP PARAMETER           COMSIDP     699    A
COMSIDP  *           LIST. (DESIRED WORD IS AT-- FWA OF PARAMETER LIST + SN=XXX)         COMSIDP     700    A
COMSIDP                                                                                  COMSIDP     701    A
COMSIDP                                                                                  COMSIDP     702    A
COMSIDP   SN=FRK   =      0           1ST WORD OF FREQUENCY PARAMETER LIST               COMSIDP     703    A
COMSIDP   SN=LL    =      0           LOWER LIMIT                                        COMSIDP     704    A
COMSIDP   SN=UL    =      1           UPPER LIMIT                                        COMSIDP     705    A
COMSIDP   SN=INC   =      2           INCREMENT                                          COMSIDP     706    A
COMSIDP   SN=HDR   =      3           HEADER                                             COMSIDP     707    A
COMSIDP   SN=CNT   =      4           SNAP COUNT                                         COMSIDP     708    A
COMSIDP   SN=FWA   =      5           FWA                                                COMSIDP     709    A
COMSIDP   SN=LWA   =      6           LWA                                                COMSIDP     710    A
COMSIDP   SN=LEN   =      7           LENGTH                                             COMSIDP     711    A
COMSIDP   SN=USF   =      8           USER *SNAP* MACRO FLAGS                            COMSIDP     712    A
COMSIDP   SN=SRL   =      9           1ST WORD OF *RGR=* LIST FOR *SNAP* MACRO           COMSIDP     713    A
COMSIDP                                                                                  COMSIDP     714    A
COMSIDP   SN=URF   =      5           USER *REG* MACRO FLAGS                             COMSIDP     715    A
COMSIDP   SN=RRL   =      6           1ST WORD OF *RGR=* LIST FOR *REG* MACRO            COMSIDP     716    A
COMSIDP                                                                                  COMSIDP     717    A
COMSIDP   SN=BRF   =      5           USER *BREAK* MACRO FLAGS                           COMSIDP     718    A
COMSIDP   SN=BRAD  =      6           USER-SPECIFIED *BREAK* ADDR                        COMSIDP     719    A
COMSIDP   O.       SPACE  4,10                                                           COMSIDP     720    A
COMSIDP  **        O. - DEFINE TOKEN (OPERATOR/OPERAND) TYPES.                           COMSIDP     721    A
COMSIDP                                                                                  COMSIDP     722    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMSIDP

```
COMSIDP                                                                                COMSIDP      723     A
COMSIDP   BOS      DEFTOT                                                              COMSIDP      724     A
COMSIDP   O.EOS    =      O.BOS                                                        COMSIDP      725     A
COMSIDP   O.NUL    =      O.BOS                                                        COMSIDP      726     A
COMSIDP   KEY      DEFTOT                                                              COMSIDP      727     A
COMSIDP   VAR      DEFTOT                                                              COMSIDP      728     A
COMSIDP   CONS     DEFTOT                                                              COMSIDP      729     A
COMSIDP   REG      DEFTOT                                                              COMSIDP      730     A
COMSIDP                                                                                COMSIDP      731     A
COMSIDP   PLUS     DEFTOT            +                                                 COMSIDP      732     A
COMSIDP   MIN      DEFTOT            -                                                 COMSIDP      733     A
COMSIDP   STAR     DEFTOT            *                                                 COMSIDP      734     A
COMSIDP   LP       DEFTOT            (                                                 COMSIDP      735     A
COMSIDP   RP       DEFTOT            )                                                 COMSIDP      736     A
COMSIDP   SHFT     DEFTOT            $                                                 COMSIDP      737     A
COMSIDP   AND      DEFTOT            .AND.                                             COMSIDP      738     A
COMSIDP   OR       DEFTOT            .OR.                                              COMSIDP      739     A
COMSIDP   XOR      DEFTOT            .XOR.                                             COMSIDP      740     A
COMSIDP   NOT      DEFTOT            .NOT.                                             COMSIDP      741     A
COMSIDP   COM      DEFTOT            ,                                                 COMSIDP      742     A
COMSIDP   PER      DEFTOT            .                                                 COMSIDP      743     A
COMSIDP   SQOT     DEFTOT            '                                                 COMSIDP      744     A
COMSIDP   QQQ      DEFTOT            ?                                                 COMSIDP      745     A
COMSIDP   ILL      DEFTOT                                                              COMSIDP      746     A
COMSIDP   O.SEP    =      O.PLUS                                                       COMSIDP      747     A
COMSIDP   BB.      SPACE  4,10                                                         COMSIDP      748     A
COMSIDP  **        BB. - *COMCBUB* STRUCTURE DEFINITIONS.                              COMSIDP      749     A
COMSIDP  *                                                                             COMSIDP      750     A
COMSIDP  *                                                                             COMSIDP      751     A
COMSIDP                                                                                COMSIDP      752     A
COMSIDP                                                                                COMSIDP      753     A
COMSIDP            DESCRIBE BB.                                                        COMSIDP      754     A
COMSIDP   TOC      DEFINE 42                                                           COMSIDP      755     A
COMSIDP   TOT      DEFINE 18                                                           COMSIDP      756     A
COMSIDP   BN.      SPACE  4,10                                                         COMSIDP      757     A
COMSIDP  **        BN. - *COMCBUN* STRUCTURE DEFINITIONS.                              COMSIDP      758     A
COMSIDP  *                                                                             COMSIDP      759     A
COMSIDP  *                                                                             COMSIDP      760     A
COMSIDP                                                                                COMSIDP      761     A
COMSIDP                                                                                COMSIDP      762     A
COMSIDP            DESCRIBE BN.                                                        COMSIDP      763     A
COMSIDP   TOC      DEFINE 42                                                           COMSIDP      764     A
COMSIDP   TOT      DEFINE 18                                                           COMSIDP      765     A
COMSIDP   BTC/XTC  SPACE  4,10                                                         COMSIDP      766     A
COMSIDP  **        BTC/XTC - *BREAK/STEP* TYPE CODES.                                  COMSIDP      767     A
COMSIDP  *                                                                             COMSIDP      768     A
COMSIDP  *                                                                             COMSIDP      769     A
COMSIDP                                                                                COMSIDP      770     A
COMSIDP                                                                                COMSIDP      771     A
COMSIDP   BTC.PL   =      1          BRPL                                              COMSIDP      772     A
COMSIDP   XTC.PL   =      1          STPL                                              COMSIDP      773     A
COMSIDP                                                                                COMSIDP      774     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMSIDP

```
COMSIDP   BTC.MI   =      2          BRMI                                          COMSIDP     775     A
COMSIDP   XTC.MI   =      2          STMI                                          COMSIDP     776     A
COMSIDP                                                                            COMSIDP     777     A
COMSIDP   BTC.ZR   =      3          BRZR                                          COMSIDP     778     A
COMSIDP   XTC.ZR   =      3          STZR                                          COMSIDP     779     A
COMSIDP                                                                            COMSIDP     780     A
COMSIDP   BTC.NZ   =      4          BRNZ                                          COMSIDP     781     A
COMSIDP   XTC.NZ   =      4          STNZ                                          COMSIDP     782     A
COMSIDP                                                                            COMSIDP     783     A
COMSIDP   BTC.EQ   =      5          BREQ                                          COMSIDP     784     A
COMSIDP   XTC.EQ   =      5          STEQ                                          COMSIDP     785     A
COMSIDP                                                                            COMSIDP     786     A
COMSIDP   BTC.NE   =      6          BRNE                                          COMSIDP     787     A
COMSIDP   XTC.NE   =      6          STNE                                          COMSIDP     788     A
COMSIDP                                                                            COMSIDP     789     A
COMSIDP   BTC.LT   =      7          BRLT                                          COMSIDP     790     A
COMSIDP   XTC.LT   =      7          STLT                                          COMSIDP     791     A
COMSIDP                                                                            COMSIDP     792     A
COMSIDP   BTC.GE   =      8          BRGE                                          COMSIDP     793     A
COMSIDP   XTC.GE   =      8          STGE                                          COMSIDP     794     A
COMSIDP                                                                            COMSIDP     795     A
COMSIDP   BTC.LE   =      9          BRLE                                          COMSIDP     796     A
COMSIDP   XTC.LE   =      9          STLE                                          COMSIDP     797     A
COMSIDP                                                                            COMSIDP     798     A
COMSIDP   BTC.GT   =      10         BRGT                                          COMSIDP     799     A
COMSIDP   XTC.GT   =      10         STGT                                          COMSIDP     800     A
COMSIDP                                                                            COMSIDP     801     A
COMSIDP   XTC.RNG  =      11         STRANGE                                       COMSIDP     802     A
COMSIDP   XTC.NR   =      12         STNR                                          COMSIDP     803     A
COMSIDP   XTC.AR   =      13         STAR                                          COMSIDP     804     A
COMSIDP   IDF.     SPACE  4,8                                                      COMSIDP     805     A
COMSIDP  **        IDF. - *IDPFLG* STRUCTURE DEFINITIONS.                          COMSIDP     806     A
COMSIDP                                                                            COMSIDP     807     A
COMSIDP                                                                            COMSIDP     808     A
COMSIDP            DESCRIBE IDF.                                                   COMSIDP     809     A
COMSIDP   IDO      DEFINE 1                                                        COMSIDP     810     A
COMSIDP   IKO      DEFINE 1                                                        COMSIDP     811     A
COMSIDP            DEFINE 1                                                        COMSIDP     812     A
COMSIDP   BDO      DEFINE 1                                                        COMSIDP     813     A
COMSIDP   BKO      DEFINE 1                                                        COMSIDP     814     A
COMSIDP            DEFINE 1                                                        COMSIDP     815     A
COMSIDP   FTO      DEFINE 1                                                        COMSIDP     816     A
COMSIDP   INP      DEFINE 1                                                        COMSIDP     817     A
COMSIDP   ADR      DEFINE 1                                                        COMSIDP     818     A
COMSIDP   XEC      DEFINE 3                                                        COMSIDP     819     A
COMSIDP   BRF      DEFINE 1                                                        COMSIDP     820     A
COMSIDP   MEM      DEFINE 1                                                        COMSIDP     821     A
COMSIDP   RAP      DEFINE 1                                                        COMSIDP     822     A
COMSIDP   XJR      DEFINE 1                                                        COMSIDP     823     A
COMSIDP            DEFINE 26                                                       COMSIDP     824     A
COMSIDP   SNL      DEFINE 18                                                       COMSIDP     825     A
COMSIDP   KW.      SPACE  4,10                                                     COMSIDP     826     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMSIDP

```
COMSIDP  **        *KEYW* TABLE STRUCTURE DEFINITIONS.                    COMSIDP    827    A
COMSIDP                                                                   COMSIDP    828    A
COMSIDP                                                                   COMSIDP    829    A
COMSIDP            DESCRIBE KW.                                           COMSIDP    830    A
COMSIDP   PNR      DEFINE 6                                               COMSIDP    831    A
COMSIDP   PAD      DEFINE 18                                              COMSIDP    832    A
COMSIDP   XAT      DEFINE 9                                               COMSIDP    833    A
COMSIDP   LEN      DEFINE 9                                               COMSIDP    834    A
COMSIDP   KEY      DEFINE 18                                              COMSIDP    835    A
COMSIDP                                                                   COMSIDP    836    A
COMSIDP            REDEF                                                  COMSIDP    837    A
COMSIDP   ATTR     DEFINE 24                                              COMSIDP    838    A
COMSIDP   LAD      DEFINE 9                                               COMSIDP    839    A
COMSIDP   UDO      DEFINE 9                                               COMSIDP    840    A
COMSIDP   HAD      DEFINE 18                                              COMSIDP    841    A
COMSIDP   PAF.     SPACE  4,8                                             COMSIDP    842    A
COMSIDP  **        PAF. - *PATFLG* STRUCTURE DEFINITIONS.                 COMSIDP    843    A
COMSIDP                                                                   COMSIDP    844    A
COMSIDP                                                                   COMSIDP    845    A
COMSIDP            DESCRIBE PAF.,15                                       COMSIDP    846    A
COMSIDP   LWA      DEFINE 1                                               COMSIDP    847    A
COMSIDP            DEFINE 2                                               COMSIDP    848    A
COMSIDP   LEN      DEFINE 1                                               COMSIDP    849    A
COMSIDP            DEFINE 11                                              COMSIDP    850    A
COMSIDP   PS.      SPACE  4,10                                            COMSIDP    851    A
COMSIDP  **        PS. - *POT* STRUCTURE DEFINITIONS.                     COMSIDP    852    A
COMSIDP                                                                   COMSIDP    853    A
COMSIDP                                                                   COMSIDP    854    A
COMSIDP            DESCRIBE PS.                                           COMSIDP    855    A
COMSIDP   PRI      DEFINE 6                                               COMSIDP    856    A
COMSIDP   UPIP     DEFINE 18                                              COMSIDP    857    A
COMSIDP   PIP      DEFINE 18                                              COMSIDP    858    A
COMSIDP   POP      DEFINE 18                                              COMSIDP    859    A
COMSIDP   TOT      DEQU   POP                                             COMSIDP    860    A
COMSIDP   PT.      SPACE  4,10                                            COMSIDP    861    A
COMSIDP  **        PT. - *PTR* (POINTER) TABLE STRUCTURE DEFINITIONS.     COMSIDP    862    A
COMSIDP                                                                   COMSIDP    863    A
COMSIDP                                                                   COMSIDP    864    A
COMSIDP            DESCRIBE PT.                                           COMSIDP    865    A
COMSIDP   FWA      DEFINE 30                                              COMSIDP    866    A
COMSIDP   LEN      DEFINE 30                                              COMSIDP    867    A
COMSIDP                                                                   COMSIDP    868    A
COMSIDP   L1.PTR   =      2          SELECT 1ST AND 2ND DEFAULTS ONLY     COMSIDP    869    A
COMSIDP   PTR.     SPACE  4,10                                            COMSIDP    870    A
COMSIDP  **        PTR. - DEFINE *PTR* ORDINALS.                          COMSIDP    871    A
COMSIDP  *                                                                COMSIDP    872    A
COMSIDP  *                                                                COMSIDP    873    A
COMSIDP                                                                   COMSIDP    874    A
COMSIDP                                                                   COMSIDP    875    A
COMSIDP   PTR.COD  CEQU   1                                               COMSIDP    876    A
COMSIDP   PTR.DPC  CEQU   3                                               COMSIDP    877    A
COMSIDP   PTR.SNP  CEQU   3                                               COMSIDP    878    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMSIDP

```
COMSIDP   PTR.STO  CEQU   3                                                    COMSIDP      879     A
COMSIDP   PTR.WHR  CEQU   1                                                    COMSIDP      880     A
COMSIDP   PTR.BRK  CEQU   1                                                    COMSIDP      881     A
COMSIDP   SC.      SPACE  4,10                                                 COMSIDP      882     A
COMSIDP  **        SC. - *STMT CONTROL* TABLE STRUCTURE DEFINITIONS.           COMSIDP      883     A
COMSIDP                                                                        COMSIDP      884     A
COMSIDP                                                                        COMSIDP      885     A
COMSIDP            DESCRIBE SC.                                                COMSIDP      886     A
COMSIDP   TOT      DEFINE 18                                                   COMSIDP      887     A
COMSIDP            DEFINE 24                                                   COMSIDP      888     A
COMSIDP   PAD      DEFINE 18                                                   COMSIDP      889     A
COMSIDP   SY.      SPACE  4,10                                                 COMSIDP      890     A
COMSIDP  **        SY. - SYMBOL TYPES.                                         COMSIDP      891     A
COMSIDP                                                                        COMSIDP      892     A
COMSIDP                                                                        COMSIDP      893     A
COMSIDP   SY.NONE  =      0           UNDEFINED                                COMSIDP      894     A
COMSIDP   SY.DECK  =      1           *FAA/RPV=RNA* DECK NAME                  COMSIDP      895     A
COMSIDP   SY.SET   =      2           *SET* NAME                               COMSIDP      896     A
COMSIDP   SY.LFN   =      3           LFN/FET NAME                             COMSIDP      897     A
COMSIDP   SY.USY   =      4           USER-DEFINED SYMBOL (VIA *USY=*)         COMSIDP      898     A
COMSIDP   TB.      SPACE  4,8                                                  COMSIDP      899     A
COMSIDP  **        TB. - TOKEN BUFFER STRUCTURE DEFINITIONS.                   COMSIDP      900     A
COMSIDP                                                                        COMSIDP      901     A
COMSIDP                                                                        COMSIDP      902     A
COMSIDP            DESCRIBE TB.                                                COMSIDP      903     A
COMSIDP   LAS      DEFINE 1                                                    COMSIDP      904     A
COMSIDP            DEFINE 23                                                   COMSIDP      905     A
COMSIDP   PRS      DEFINE 18                                                   COMSIDP      906     A
COMSIDP   LEN      DEFINE 18                                                   COMSIDP      907     A
COMSIDP   TB.      SPACE  4,10                                                 COMSIDP      908     A
COMSIDP            REDEF                                                       COMSIDP      909     A
COMSIDP   TOC      DEFINE BB.TOCL                                              COMSIDP      910     A
COMSIDP   TOT      DEFINE BB.TOTL                                              COMSIDP      911     A
COMSIDP   BB.      SPACE  4,10                                                 COMSIDP      912     A
COMSIDP            REDEF                                                       COMSIDP      913     A
COMSIDP            DEFINE 24                                                   COMSIDP      914     A
COMSIDP   0TR      DEFINE 18                                                   COMSIDP      915     A
COMSIDP   TOT      DEFINE 18                                                   COMSIDP      916     A
COMSIDP   COMSIDP  SPACE  4,8                                                  COMSIDP      917     A
COMSIDP            BASE   *                                                    COMSIDP      918     A
```

SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - COMSIDP

IDENTIFIER	TOTAL	ACTIVE
COMSIDP	918	918

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMSTOK

```
COMSTOK  *COMDECK COMSTOK                                                      COMSTOK        1     A
COMSTOK   COMSTOK  CTEXT  COMSTOK 2.0 - COMMON TOKEN GENERATOR INTERFACE TEXT. COMSTOK        2     A
COMSTOK   TOK      SPACE  4                                                    COMSTOK        3     A
```

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMSTOK

```
COMSTOK            BASE   D                                                    COMSTOK      4    A
COMSTOK   TOK      SPACE  4,10                                                 COMSTOK      5    A
COMSTOK  ***       COMSTOK - COMMON TOKEN GENERATOR INTERFACE TEXT.            COMSTOK      6    A
COMSTOK  *         P CLARE   77/02/26                                          COMSTOK      7    A
COMSTOK  *         P CLARE   78/04/27 VERSION 2.0                              COMSTOK      8    A
COMSTOK   TOK      SPACE  4,10                                                 COMSTOK      9    A
COMSTOK  ***       COMSTOK - COMMON TOKEN GENERATOR INTERFACE TEXT.            COMSTOK     10    A
COMSTOK  *                                                                     COMSTOK     11    A
COMSTOK  *                                                                     COMSTOK     12    A
COMSTOK  *         THIS COMDECK CONTAINS SYMBOL DEFINITIONS THAT ARE REQUIRED  COMSTOK     13    A
COMSTOK  *         BY THE COMMON TOKEN GENERATOR (COMCTOK) AND ITS SUPPORTING  COMSTOK     14    A
COMSTOK  *         COMDECKS.                                                   COMSTOK     15    A
COMSTOK   DEFS     SPACE  4,10                                                 COMSTOK     16    A
COMSTOK  **        SYMBOL DEFINITIONS.                                         COMSTOK     17    A
COMSTOK                                                                        COMSTOK     18    A
COMSTOK                                                                        COMSTOK     19    A
COMSTOK            IF     -DEF,BICH,1                                          COMSTOK     20    A
COMSTOK   BICH     =      1RM                                                  COMSTOK     21    A
COMSTOK                                                                        COMSTOK     22    A
COMSTOK            IF     -MIC,BLOWUP,1                                        COMSTOK     23    A
COMSTOK   BLOWUP   MICRO  1,,/*O+4S15/                                          COMSTOK     24    A
COMSTOK                                                                        COMSTOK     25    A
COMSTOK            IF     -DEF,CHAR,1                                          COMSTOK     26    A
COMSTOK   CHAR     =      6                                                    COMSTOK     27    A
COMSTOK                                                                        COMSTOK     28    A
COMSTOK            IF     -DEF,FW.TOK,1                                        COMSTOK     29    A
COMSTOK   FW.TOK   =      0                                                    COMSTOK     30    A
COMSTOK                                                                        COMSTOK     31    A
COMSTOK            IF     -MIC,O.,1                                            COMSTOK     32    A
COMSTOK   O.       MICRO  1,,/O./                                              COMSTOK     33    A
COMSTOK                                                                        COMSTOK     34    A
COMSTOK            IF     -DEF,"O."BOS,1                                       COMSTOK     35    A
COMSTOK   "O."BOS  =      0                                                    COMSTOK     36    A
COMSTOK                                                                        COMSTOK     37    A
COMSTOK            IF     -DEF,"O."EOS,1                                       COMSTOK     38    A
COMSTOK   "O."EOS  =      0                                                    COMSTOK     39    A
COMSTOK                                                                        COMSTOK     40    A
COMSTOK            IF     -MIC,O=,1                                            COMSTOK     41    A
COMSTOK   O=       MICRO  1,,/O=/                                              COMSTOK     42    A
COMSTOK                                                                        COMSTOK     43    A
COMSTOK            IF     -DEF,TEST,1                                          COMSTOK     44    A
COMSTOK   TEST     =      0                                                    COMSTOK     45    A
COMSTOK   TC.      SPACE  4,10                                                 COMSTOK     46    A
COMSTOK  **        TC. - *TOKCOM* LOCATION ORDINAL DEFINITIONS.                COMSTOK     47    A
COMSTOK  *                                                                     COMSTOK     48    A
COMSTOK  *                                                                     COMSTOK     49    A
COMSTOK  *         SYMBOLS OF THE FORM *TC.XXX*, WHERE *TC.* IS A COMMON       COMSTOK     50    A
COMSTOK  *         PREFIX AND *XXX* IS A UNIQUE NAME, DESCRIBE THE ORDINALS    COMSTOK     51    A
COMSTOK  *         INTO *TOKCOM*, THE *USER/TOK* COMMUNICATIONS AREA.          COMSTOK     52    A
COMSTOK  *                                                                     COMSTOK     53    A
COMSTOK  *         *TOKCOM* CONTAINS INFORMATION THAT IS PASSED BACK AND FORTH COMSTOK     54    A
COMSTOK  *         BETWEEN A *TOK* CALLER/USER AND *TOK* (COMMON TOKEN GENERATOR) COMSTOK     55    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMSTOK

```
COMSTOK  *         AND CONTAINS INFORMATION ABOUT THE CURRENT LINE TO BE,          COMSTOK    56   A
COMSTOK  *         OR BEING, ENTOKENED.  SEE *COMCTOK* ENTRY/EXIT PREAMBLE FOR     COMSTOK    57   A
COMSTOK  *         DETAILS.                                                        COMSTOK    58   A
COMSTOK                                                                            COMSTOK    59   A
COMSTOK                                                                            COMSTOK    60   A
COMSTOK            LOC    0                                                        COMSTOK    61   A
COMSTOK   TC.FWA   =      *                                                        COMSTOK    62   A
COMSTOK   TC.FCP   =      *+1                                                      COMSTOK    63   A
COMSTOK   TC.LWA   =      *+2                                                      COMSTOK    64   A
COMSTOK   TC.LCP   =      *+3                                                      COMSTOK    65   A
COMSTOK   TC.P     =      *+4                                                      COMSTOK    66   A
COMSTOK   TC.SOA   =      *+5                                                      COMSTOK    67   A
COMSTOK   TC.SOX   =      *+6                                                      COMSTOK    68   A
COMSTOK   TC.SOL   =      *+7                                                      COMSTOK    69   A
COMSTOK   TC.SOB   =      *+8D                                                     COMSTOK    70   A
COMSTOK   TC.SOC   =      *+9D                                                     COMSTOK    71   A
COMSTOK   TC.TBA   =      *+10D                                                    COMSTOK    72   A
COMSTOK   TC.TBB   =      *+11D                                                    COMSTOK    73   A
COMSTOK   TC.RSF   =      *+12D                                                    COMSTOK    74   A
COMSTOK   TC.BOL   =      *+13D                                                    COMSTOK    75   A
COMSTOK   TC.EOL   =      *+14D                                                    COMSTOK    76   A
COMSTOK   TC.REST  =      *+15D                                                    COMSTOK    77   A
COMSTOK            LOC    *O                                                       COMSTOK    78   A
COMSTOK                                                                            COMSTOK    79   A
COMSTOK   L.TCOM   =      16D         LEN OF *TOKCOM*                              COMSTOK    80   A
COMSTOK   TG.      SPACE  4,10                                                     COMSTOK    81   A
COMSTOK  **        TG. - TOGEL BINARY INSTRUCTION SYMBOL DEFINITIONS.              COMSTOK    82   A
COMSTOK                                                                            COMSTOK    83   A
COMSTOK                                                                            COMSTOK    84   A
COMSTOK            DESCRIBE TG.                                                    COMSTOK    85   A
COMSTOK            DEFINE 42                                                       COMSTOK    86   A
COMSTOK   TOFU     DEFINE 18                                                       COMSTOK    87   A
COMSTOK   CALT     SPACE  4,10                                                     COMSTOK    88   A
COMSTOK  **        *CALT XX* BINARY TOGEL INSTRUCTION FORMAT.                      COMSTOK    89   A
COMSTOK                                                                            COMSTOK    90   A
COMSTOK                                                                            COMSTOK    91   A
COMSTOK            REDEF                                                           COMSTOK    92   A
COMSTOK            DEFINE 24                                                       COMSTOK    93   A
COMSTOK   CAD      DEFINE 18                                                       COMSTOK    94   A
COMSTOK   TOFU     DEFINE 18                                                       COMSTOK    95   A
COMSTOK   CASEOF   SPACE  4,10                                                     COMSTOK    96   A
COMSTOK  **        *CASEOF (..)* BINARY TOGEL INSTRUCTION FORMAT.                  COMSTOK    97   A
COMSTOK                                                                            COMSTOK    98   A
COMSTOK                                                                            COMSTOK    99   A
COMSTOK            REDEF                                                           COMSTOK   100   A
COMSTOK   COA      DEFINE 12                                                       COMSTOK   101   A
COMSTOK   COZ      DEFINE 12                                                       COMSTOK   102   A
COMSTOK   TAD      DEFINE 18                                                       COMSTOK   103   A
COMSTOK   TOFU     DEFINE 18                                                       COMSTOK   104   A
COMSTOK   ENDT     SPACE  4,10                                                     COMSTOK   105   A
COMSTOK  **        *ENDT* BINARY TOGEL INSTRUCTION FORMAT.                         COMSTOK   106   A
COMSTOK                                                                            COMSTOK   107   A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMSTOK

```
COMSTOK                                                                  COMSTOK      108     A
COMSTOK            REDEF                                                 COMSTOK      109     A
COMSTOK   IFL      DEFINE 42                                             COMSTOK      110     A
COMSTOK   TOFU     DEFINE 18                                             COMSTOK      111     A
COMSTOK   IFT      SPACE  4,10                                           COMSTOK      112     A
COMSTOK  **        *IFT (...)* BINARY TOGEL INSTRUCTION FORMAT.          COMSTOK      113     A
COMSTOK                                                                  COMSTOK      114     A
COMSTOK                                                                  COMSTOK      115     A
COMSTOK            REDEF                                                 COMSTOK      116     A
COMSTOK            DEFINE 6                                              COMSTOK      117     A
COMSTOK   FAL      DEFINE 18                                             COMSTOK      118     A
COMSTOK   MXA      DEFINE 18                                             COMSTOK      119     A
COMSTOK   TOFU     DEFINE 18                                             COMSTOK      120     A
COMSTOK   GENTOK   SPACE  4,10                                           COMSTOK      121     A
COMSTOK  **        *GENTOK* BINARY TOGEL INSTRUCTION FORMAT.             COMSTOK      122     A
COMSTOK                                                                  COMSTOK      123     A
COMSTOK                                                                  COMSTOK      124     A
COMSTOK            REDEF                                                 COMSTOK      125     A
COMSTOK            DEFINE 24                                             COMSTOK      126     A
COMSTOK   GTT      DEFINE 18                                             COMSTOK      127     A
COMSTOK   TOFU     DEFINE 18                                             COMSTOK      128     A
COMSTOK   GOTO     SPACE  4,10                                           COMSTOK      129     A
COMSTOK  **        *GOTO XX* BINARY TOGEL INSTRUCTION FORMAT.            COMSTOK      130     A
COMSTOK                                                                  COMSTOK      131     A
COMSTOK                                                                  COMSTOK      132     A
COMSTOK            REDEF                                                 COMSTOK      133     A
COMSTOK            DEFINE 24                                             COMSTOK      134     A
COMSTOK   GOT      DEFINE 18                                             COMSTOK      135     A
COMSTOK   TOFU     DEFINE 18                                             COMSTOK      136     A
COMSTOK   GROUP    SPACE  4,10                                           COMSTOK      137     A
COMSTOK  **        *GROUP* BINARY TOGEL INSTRUCTION SYMBOL DEFS.         COMSTOK      138     A
COMSTOK                                                                  COMSTOK      139     A
COMSTOK                                                                  COMSTOK      140     A
COMSTOK            REDEF                                                 COMSTOK      141     A
COMSTOK            DEFINE 6                                              COMSTOK      142     A
COMSTOK   TOT      DEFINE 18                                             COMSTOK      143     A
COMSTOK   MXA      DEFINE 18                                             COMSTOK      144     A
COMSTOK   TOFU     DEFINE 18                                             COMSTOK      145     A
COMSTOK   ON       SPACE  4,10                                           COMSTOK      146     A
COMSTOK  **        *ON I,V* BINARY TOGEL INSTRUCTION FORMAT.             COMSTOK      147     A
COMSTOK                                                                  COMSTOK      148     A
COMSTOK                                                                  COMSTOK      149     A
COMSTOK            REDEF                                                 COMSTOK      150     A
COMSTOK            DEFINE 6                                              COMSTOK      151     A
COMSTOK   INV      DEFINE 18                                             COMSTOK      152     A
COMSTOK   INC      DEFINE 18                                             COMSTOK      153     A
COMSTOK   TOFU     DEFINE 18                                             COMSTOK      154     A
COMSTOK   XEQ      SPACE  4,10                                           COMSTOK      155     A
COMSTOK  **        *XEQ P* BINARY TOGEL INSTRUCTION FORMAT.              COMSTOK      156     A
COMSTOK                                                                  COMSTOK      157     A
COMSTOK                                                                  COMSTOK      158     A
COMSTOK            REDEF                                                 COMSTOK      159     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN COMSTOK

```
COMSTOK            DEFINE 24                                                COMSTOK      160     A
COMSTOK   XAD      DEFINE 18                                                COMSTOK      161     A
COMSTOK   TOFU     DEFINE 18                                                COMSTOK      162     A
COMSTOK   TK.      SPACE  4,10                                              COMSTOK      163     A
COMSTOK  **        TK. - TOKEN FORMAT SYMBOL DEFINITIONS.                   COMSTOK      164     A
COMSTOK  *                                                                  COMSTOK      165     A
COMSTOK  *                                                                  COMSTOK      166     A
COMSTOK   TK.      SPACE  4,10                                              COMSTOK      167     A
COMSTOK  **         *TOKEN* FORMAT.                                         COMSTOK      168     A
COMSTOK                                                                     COMSTOK      169     A
COMSTOK                                                                     COMSTOK      170     A
COMSTOK            DESCRIBE TK.                                             COMSTOK      171     A
COMSTOK            DEFINE 6                                                 COMSTOK      172     A
COMSTOK   PAD      DEFINE 18                                                COMSTOK      173     A
COMSTOK   LNK      DEFINE 18                                                COMSTOK      174     A
COMSTOK   TOT      DEFINE 18                                                COMSTOK      175     A
COMSTOK   TL.      SPACE  4,10                                              COMSTOK      176     A
COMSTOK  **        *TLINK* FORMAT.                                          COMSTOK      177     A
COMSTOK                                                                     COMSTOK      178     A
COMSTOK                                                                     COMSTOK      179     A
COMSTOK            DESCRIBE TL.                                             COMSTOK      180     A
COMSTOK   MAF      DEFINE 1                                                 COMSTOK      181     A
COMSTOK            DEFINE 5                                                 COMSTOK      182     A
COMSTOK   TRU      DEFINE 18                                                COMSTOK      183     A
COMSTOK   FAL      DEFINE 18                                                COMSTOK      184     A
COMSTOK   PRE      DEFINE 18                                                COMSTOK      185     A
COMSTOK   TL.      SPACE  4,10                                              COMSTOK      186     A
COMSTOK            REDEF                                                    COMSTOK      187     A
COMSTOK   MAC      DEFINE 42                                                COMSTOK      188     A
COMSTOK            DEFINE 18                                                COMSTOK      189     A
COMSTOK   TL.      SPACE  4,10                                              COMSTOK      190     A
COMSTOK            REDEF                                                    COMSTOK      191     A
COMSTOK            DEFINE 6                                                 COMSTOK      192     A
COMSTOK   PAD      DEFINE TK.PADL                                           COMSTOK      193     A
COMSTOK   LNK      DEFINE TK.LNKL                                           COMSTOK      194     A
COMSTOK   TOT      DEFINE TK.TOTL                                           COMSTOK      195     A
COMSTOK   TOK      SPACE  4,10                                              COMSTOK      196     A
COMSTOK   COMSTOK  ENDX                                                     COMSTOK      197     A
```

SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - COMSTOK

IDENTIFIER	TOTAL	ACTIVE
COMSTOK	197	197

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCOMRPV

```
CCOMRPV  *COMDECK  CCOMRPV                 COMPILE TIME REPRIEVE PROCESSOR.     000140   CCOMRPV        1     A
CCOMRPV            CTEXT  CCOMRPV - COMPILE TIME REPRIEVE PROCESSOR.            000150   CCOMRPV        2     A
CCOMRPV   RPV=     SPACE  4,10                                                  000160   CCOMRPV        3     A
CCOMRPV            IF     -DEF,QUAL$,1                                          000170   CCOMRPV        4     A
CCOMRPV            QUAL   CCOMRPV                                               000180   CCOMRPV        5     A
```

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCOMRPV

```
CCOMRPV   RPV=     SPACE  4,10                                          000190   CCOMRPV        6     A
CCOMRPV  **        RPV= - REPRIEVE PROCESSOR.                           000200   CCOMRPV        7     A
CCOMRPV  *                                                              000210   CCOMRPV        8     A
CCOMRPV  *         IF THE PROGRAM IS INTERRUPTED BY THE SYSTEM, *RPV=* GAINS     000220   CCOMRPV        9     A
CCOMRPV  *         CONTROL TO ISSUE DAYFILE MESSAGES THAT IDENTIFY THE PROGRAM   000230   CCOMRPV       10     A
CCOMRPV  *         UNIT BEING COMPILED AND WHERE THE ERROR OCCURED.  A USER      000240   CCOMRPV       11     A
CCOMRPV  *         ROUTINE IS CALLED FOR POST PROCESSING AND ALL FILES IN AN     000250   CCOMRPV       12     A
CCOMRPV  *         OUTPUT MODE ARE FLUSHED.  FINALLY, THE ORIGINAL ERROR         000260   CCOMRPV       13     A
CCOMRPV  *         CONDITION IS RESTORED TO PERMIT NORMAL *EXIT*                 000270   CCOMRPV       14     A
CCOMRPV  *         CONDITION PROCESSING.                                         000280   CCOMRPV       15     A
CCOMRPV  *                                                              000290   CCOMRPV       16     A
CCOMRPV  *         ENTRY  (RPV=CLN) = CURRENT LINE NUMBER IN BINARY              000300   CCOMRPV       17     A
CCOMRPV  *                            IF DEF, LINE NUMBER MESSAGE ISSUED         000310   CCOMRPV       18     A
CCOMRPV  *                (RPV=FVT) = FIRST ENTRY OF A FILE VECTOR --            000320   CCOMRPV       19     A
CCOMRPV  *                            42/0LLFN, 18/FET ADDRESS                   000330   CCOMRPV       20     A
CCOMRPV  *                            TERMINATED BY A -1 WORD                    000340   CCOMRPV       21     A
CCOMRPV  *                            IF DEF, OUTPUT FILES ARE FLUSHED           000350   CCOMRPV       22     A
CCOMRPV  *                (RPV=LOL) = LAST OVERLAY LOADED                        000360   CCOMRPV       23     A
CCOMRPV  *                            6/PRIMARY,6/SECONDARY,48/UNUSED            000370   CCOMRPV       24     A
CCOMRPV  *                            IF DEF, OVERLAY LOADED MESSAGE ISSUED      000380   CCOMRPV       25     A
CCOMRPV  *                (RPV=MSG) = IF DEF AND .NE. 0, FIRST WORD OF DAYFILE   000390   CCOMRPV       26     A
CCOMRPV  *                             MESSAGE (=C FORMAT) TO BE ISSUED BY RPV=. 000400   CCOMRPV       27     A
CCOMRPV  *                (RPV=RNA) = FIRST WORD OF A ROUTINE NAME/ADDRESS TABLE 000410   CCOMRPV       28     A
CCOMRPV  *                            BLOCKS OF THE FORM --                      000420   CCOMRPV       29     A
CCOMRPV  *                            42/0LNAME, 18/FWA                          000430   CCOMRPV       30     A
CCOMRPV  *                            ENDED BY -                                 000440   CCOMRPV       31     A
CCOMRPV  *                            1/1,41/UNUSED,18/NEXT BLOCK FWA OR ZERO IF 000450   CCOMRPV       32     A
CCOMRPV  *                              LAST BLOCK.                              000460   CCOMRPV       33     A
CCOMRPV  *                            IF -DEF, ABS ADDRESS IS USED.              000470   CCOMRPV       34     A
CCOMRPV  *                (RPV=URP) = ADDRESS OF USER REPRIEVE PROCESSING,       000480   CCOMRPV       35     A
CCOMRPV  *                            42/0,18/ADDRESS TO RJ TO                   000490   CCOMRPV       36     A
CCOMRPV  *                                                              000500   CCOMRPV       37     A
CCOMRPV  *                CP#RM = 0 FOR DIRECT CIO, 7 FOR SCOPE 2 RECORD MANGLER 000510   CCOMRPV       38     A
CCOMRPV  *                .OS = 1 FOR NOS, 2 FOR SCOPE 2, 3 FOR NOS/BE           000520   CCOMRPV       39     A
CCOMRPV  *                                                              000530   CCOMRPV       40     A
CCOMRPV  *         EXIT   MESSAGES POSTED AND BUFFERS FLUSHED.  REINSTATES THE   000540   CCOMRPV       41     A
CCOMRPV  *                ERROR CONDITION AND RETURNS CONTROL TO THE OPERATING   000550   CCOMRPV       42     A
CCOMRPV  *                SYSTEM.                                               000560   CCOMRPV       43     A
CCOMRPV  *                                                              000570   CCOMRPV       44     A
CCOMRPV  *         CALLS  CDD, CIO=, COD, FRA=, MSG=, SYS=, WNB=                 000580   CCOMRPV       45     A
CCOMRPV                                                                 000590   CCOMRPV       46     A
CCOMRPV                                                                 000600   CCOMRPV       47     A
CCOMRPV   RPV=     BSS    0           SYSTEM ENTRY AT THIS ADDRESS + 21B         000610   CCOMRPV       48     A
CCOMRPV   #OS      IFNE   .OS,2       IF NOT SCOPE 2                             000620   CCOMRPV       49     A
CCOMRPV   XJP      VFD    12/0,18/LWARPV,30/0                                    000630   CCOMRPV       50     A
CCOMRPV   #OS      ELSE                                                          000640   CCOMRPV       51     A
CCOMRPV   XJP      VFD    12/0,18/LWARPV,6/0,24/77770014B                        000650   CCOMRPV       52     A
CCOMRPV   #OS      ENDIF                                                         000660   CCOMRPV       53     A
CCOMRPV                                                                 000670   CCOMRPV       54     A
CCOMRPV            BSSZ   16          EXCHANGE PACKAGE AND RA+1                  000680   CCOMRPV       55     A
CCOMRPV   RPV=     SPACE  4,10                                           000690   CCOMRPV       56     A
CCOMRPV  **        RPV - REPRIEVE PROCESSOR SYSTEM ENTRY POINT.                  000700   CCOMRPV       57     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCOMRPV

```
CCOMRPV  *                                                          000710  CCOMRPV   58  A
CCOMRPV  *         ENTRY  (XJP TO XJP+15) = EXCHANGE PACKAGE AT TIME OF ERROR  000720  CCOMRPV   59  A
CCOMRPV  *                (XJP+16) = CONTENTS OF RA+1 AT THE TIME OF THE ERROR 000730  CCOMRPV   60  A
CCOMRPV                                                             000740  CCOMRPV   61  A
CCOMRPV   RPV      SB1    1           SYSTEM RPV ENTRY              000750  CCOMRPV   62  A
CCOMRPV                                                             000760  CCOMRPV   63  A
CCOMRPV  *         SAVE CONTENTS OF ENTRY POINTS THAT *RPV=* WILL USE. 000770  CCOMRPV   64  A
CCOMRPV                                                             000780  CCOMRPV   65  A
CCOMRPV            SA1    =XCDD                                     000790  CCOMRPV   66  A
CCOMRPV            SA2    =XCOD                                     000800  CCOMRPV   67  A
CCOMRPV            BX6    X1                                        000810  CCOMRPV   68  A
CCOMRPV            LX7    X2                                        000820  CCOMRPV   69  A
CCOMRPV            SA6    RPVE                                      000830  CCOMRPV   70  A
CCOMRPV            SA7    A6+1                                      000840  CCOMRPV   71  A
CCOMRPV                                                             000850  CCOMRPV   72  A
CCOMRPV   #OS      IFNE   .OS,2       IF NOT SCOPE 2                000860  CCOMRPV   73  A
CCOMRPV            SA1    =XSYS=                                    000870  CCOMRPV   74  A
CCOMRPV            SA2    =XWNB=                                    000880  CCOMRPV   75  A
CCOMRPV            SA3    =XMSG=                                    000890  CCOMRPV   76  A
CCOMRPV            BX6    X1                                        000900  CCOMRPV   77  A
CCOMRPV            LX7    X2                                        000910  CCOMRPV   78  A
CCOMRPV            SA6    A7+B1                                     000920  CCOMRPV   79  A
CCOMRPV            SA7    A6+B1                                     000930  CCOMRPV   80  A
CCOMRPV            BX6    X3                                        000940  CCOMRPV   81  A
CCOMRPV            SA6    A7+B1                                     000950  CCOMRPV   82  A
CCOMRPV                                                             000960  CCOMRPV   83  A
CCOMRPV   #RM      IFEQ   CP#RM,0     IF DIRECT CIO I/O             000970  CCOMRPV   84  A
CCOMRPV            SA1    =XCIO=                                    000980  CCOMRPV   85  A
CCOMRPV            BX6    X1                                        000990  CCOMRPV   86  A
CCOMRPV            SA6    A6+B1                                     001000  CCOMRPV   87  A
CCOMRPV   #RM      ENDIF                                            001010  CCOMRPV   88  A
CCOMRPV   #OS      ENDIF                                            001020  CCOMRPV   89  A
CCOMRPV                                                             001030  CCOMRPV   90  A
CCOMRPV                                                             001040  CCOMRPV   91  A
CCOMRPV  *         EXTRACT ERROR ADDRESS.                           001050  CCOMRPV   92  A
CCOMRPV                                                             001060  CCOMRPV   93  A
CCOMRPV            SA5    XJP                                       001070  CCOMRPV   94  A
CCOMRPV            MX0    -17                                       001080  CCOMRPV   95  A
CCOMRPV            LX5    24                                        001090  CCOMRPV   96  A
CCOMRPV            BX1    -X0*X5      EXTRACT P REGISTER            001100  CCOMRPV   97  A
CCOMRPV            SB2    X1                                        001110  CCOMRPV   98  A
CCOMRPV            GT     B2,B1,RPV2  IF P-REGISTER NOT CLEARED     001120  CCOMRPV   99  A
CCOMRPV            SA5    0+                                        001130  CCOMRPV  100  A
CCOMRPV            LX5    59-47+18                                  001140  CCOMRPV  101  A
CCOMRPV            BX1    -X0*X5      EXTRACT P FROM RA+0           001150  CCOMRPV  102  A
CCOMRPV   RPV2     BSS    0                                         001160  CCOMRPV  103  A
CCOMRPV                                                             001170  CCOMRPV  104  A
CCOMRPV                                                             001180  CCOMRPV  105  A
CCOMRPV  *         DETERMINE ROUTINE RELATIVE ADDRESS, IF *RNA* TABLE EXISTS. 001190  CCOMRPV  106  A
CCOMRPV                                                             001200  CCOMRPV  107  A
CCOMRPV   #RPV=    IF DEF,RPV=RNA                                   001210  CCOMRPV  108  A
CCOMRPV            SB7    RPV3        (B7) = RETURN ADDRESS         001220  CCOMRPV  109  A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCOMRPV

```
CCOMRPV            EQ     FRA=        FIND RELATIVE ADDRESS                    001230   CCOMRPV      110     A
CCOMRPV                                                                        001240   CCOMRPV      111     A
CCOMRPV   RPV3     NZ     X7,RPV3A    IF RELATIVE                              001250   CCOMRPV      112     A
CCOMRPV            MX0    42                                                   001260   CCOMRPV      113     A
CCOMRPV            BX6    X0*X4                                                001270   CCOMRPV      114     A
CCOMRPV   RPV3A    SA6    RPVC+1                                               001280   CCOMRPV      115     A
CCOMRPV            SA7    A6+1                                                 001290   CCOMRPV      116     A
CCOMRPV                                                                        001300   CCOMRPV      117     A
CCOMRPV   *        CONVERT ABS ADDRESS IF NO *RNA* TABLE.                      001310   CCOMRPV      118     A
CCOMRPV                                                                        001320   CCOMRPV      119     A
CCOMRPV   #RPV=    ELSE                                                        001330   CCOMRPV      120     A
CCOMRPV            RJ     =XCOD       CONVERT OCTAL DIGITS                     001340   CCOMRPV      121     A
CCOMRPV            MX0    42                                                   001350   CCOMRPV      122     A
CCOMRPV            BX6    X0*X4       ADD ZERO BYTE TERMINATOR                 001360   CCOMRPV      123     A
CCOMRPV            SA6    RPVC+1                                               001370   CCOMRPV      124     A
CCOMRPV   #RPV=    ENDIF                                                       001380   CCOMRPV      125     A
CCOMRPV                                                                        001390   CCOMRPV      126     A
CCOMRPV   *        DETERMINE LAST OVERLAY LOADED.                              001400   CCOMRPV      127     A
CCOMRPV                                                                        001410   CCOMRPV      128     A
CCOMRPV   #RPV=    IF     DEF,RPV=LOL                                          001420   CCOMRPV      129     A
CCOMRPV            SA1    RPV=LOL     (X1) = LAST OVERLAY 6/PRI,6/SEC,48/UNUSED 001430   CCOMRPV      130     A
CCOMRPV            MX0    -6                                                   001440   CCOMRPV      131     A
CCOMRPV            LX1    6                                                    001450   CCOMRPV      132     A
CCOMRPV            BX6    -X0*X1      PRIMARY LEVEL                            001460   CCOMRPV      133     A
CCOMRPV            LX1    6                                                    001470   CCOMRPV      134     A
CCOMRPV            BX7    -X0*X1      SECONDARY LEVEL                          001480   CCOMRPV      135     A
CCOMRPV            LX6    12                                                   001490   CCOMRPV      136     A
CCOMRPV            SB2    X7          (B2) = SECONDARY OVERLAY LEVEL NUMBER    001500   CCOMRPV      137     A
CCOMRPV            BX7    X6+X7                                                001510   CCOMRPV      138     A
CCOMRPV            SA2    RPVD+2      * - (0,0)  *                             001520   CCOMRPV      139     A
CCOMRPV            LX7    18                                                   001530   CCOMRPV      140     A
CCOMRPV            IX6    X2+X7                                                001540   CCOMRPV      141     A
CCOMRPV            SA6    A2+                                                  001550   CCOMRPV      142     A
CCOMRPV   #RPV=    ENDIF                                                       001560   CCOMRPV      143     A
CCOMRPV                                                                        001570   CCOMRPV      144     A
CCOMRPV   *        DETERMINE LAST SOURCE STATEMENT PROCESSED.                  001580   CCOMRPV      145     A
CCOMRPV                                                                        001590   CCOMRPV      146     A
CCOMRPV   #RPV=    IF     DEF,RPV=CLN                                          001600   CCOMRPV      147     A
CCOMRPV            SA1    RPV=CLN     CURRENT LINE NUMBER                      001610   CCOMRPV      148     A
CCOMRPV            BX2    X1                                                   001620   CCOMRPV      149     A
CCOMRPV            AX2    59                                                   001630   CCOMRPV      150     A
CCOMRPV            BX1    -X2*X1      INSURE POSITIVE NUMBER                   001640   CCOMRPV      151     A
CCOMRPV            RJ     =XCDD       CONVERT DECIMAL DIGITS                   001650   CCOMRPV      152     A
CCOMRPV            MX0    48                                                   001660   CCOMRPV      153     A
CCOMRPV            BX6    X0*X4       CREATE 12-BIT ZERO BYTE MSG TERMINATOR   001670   CCOMRPV      154     A
CCOMRPV            SA6    RPVB+3      LINE NUMBER TO MESSAGE TEXT              001680   CCOMRPV      155     A
CCOMRPV   #RPV=    ENDIF                                                       001690   CCOMRPV      156     A
CCOMRPV                                                                        001700   CCOMRPV      157     A
CCOMRPV   *        ISSUE THE DAYFILE MESSAGES.                                 001710   CCOMRPV      158     A
CCOMRPV                                                                        001720   CCOMRPV      159     A
CCOMRPV            IF     DEF,RPV=MSG,1                                        001730   CCOMRPV      160     A
CCOMRPV            MESSAGE  RPV=MSG,,RCL    *COMPILING NAME*                   001740   CCOMRPV      161     A
```

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCOMRPV

```
CCOMRPV                                                                  001750   CCOMRPV      162     A
CCOMRPV            IF     DEF,RPV=CLN,1                                  001760   CCOMRPV      163     A
CCOMRPV            MESSAGE   RPVB,,RCL      * LAST STATEMENT BEGAN AT LINE NNNN* 001770   CCOMRPV      164     A
CCOMRPV                                                                  001780   CCOMRPV      165     A
CCOMRPV            MESSAGE   RPVC,,RCL      * ERROR AT XXXXXX IN YYYYYYY*  001790   CCOMRPV      166     A
CCOMRPV                                                                  001800   CCOMRPV      167     A
CCOMRPV            IF     DEF,RPV=LOL,1                                  001810   CCOMRPV      168     A
CCOMRPV            MESSAGE   RPVD,,RCL      * LAST OVERLAY LOADED - (P,S)* 001820   CCOMRPV      169     A
CCOMRPV                                                                  001830   CCOMRPV      170     A
CCOMRPV  *         CALL USER REPRIEVE PROCESSOR.                         001840   CCOMRPV      171     A
CCOMRPV                                                                  001850   CCOMRPV      172     A
CCOMRPV   #RPV=    IF     DEF,RPV=URP                                    001860   CCOMRPV      173     A
CCOMRPV            SA1    RPV=URP     USER REPRIEVE PROCESSOR            001870   CCOMRPV      174     A
CCOMRPV            ZR     X1,RPV4     IF NO USER ROUTINE SPECIFIED       001880   CCOMRPV      175     A
CCOMRPV                                                                  001890   CCOMRPV      176     A
CCOMRPV            MX0    43                                             001900   CCOMRPV      177     A
CCOMRPV            BX3    X0*X1                                          001910   CCOMRPV      178     A
CCOMRPV            NZ     X3,RPV4     IF BAD ADDRESS                     001920   CCOMRPV      179     A
CCOMRPV                                                                  001930   CCOMRPV      180     A
CCOMRPV            SX2    RPV4                                           001940   CCOMRPV      181     A
CCOMRPV            SB7    X1                                             001950   CCOMRPV      182     A
CCOMRPV            LX2    32                                             001960   CCOMRPV      183     A
CCOMRPV            MI     B7,RPV4     IF BAD ADDRESS                     001970   CCOMRPV      184     A
CCOMRPV                                                                  001980   CCOMRPV      185     A
CCOMRPV            PX6    X2                                             001990   CCOMRPV      186     A
CCOMRPV            LX6    -2          (X6) = *EQ RPV4*                   002000   CCOMRPV      187     A
CCOMRPV            SA6    B7          STORE FOR USER PROCESSOR RETURN    002010   CCOMRPV      188     A
CCOMRPV            JP     B7+1        CALL USER PROCESSOR...             002020   CCOMRPV      189     A
CCOMRPV   RPV4     BSS    0           ...RETURN FROM USER PROCESSOR      002030   CCOMRPV      190     A
CCOMRPV   #RPV=    ENDIF                                                 002040   CCOMRPV      191     A
CCOMRPV                                                                  002050   CCOMRPV      192     A
CCOMRPV  *         FLUSH FILE I/O BUFFERS (DIRECT CIO I/O ONLY).         002060   CCOMRPV      193     A
CCOMRPV                                                                  002070   CCOMRPV      194     A
CCOMRPV   #RPV=    IF     DEF,RPV=FVT IF FILE VECTOR TABLE EXISTS        002080   CCOMRPV      195     A
CCOMRPV   #RM      IFEQ   CP#RM,0     IF DIRECT CIO I/O                  002090   CCOMRPV      196     A
CCOMRPV            SA0    RPV=FVT                                        002100   CCOMRPV      197     A
CCOMRPV            SB7    +                                              002110   CCOMRPV      198     A
CCOMRPV   RPV5     SA2    A0+B7       (X2) = NEXT FILE FET ADDRESS       002120   CCOMRPV      199     A
CCOMRPV            SB6    X2+B1                                          002130   CCOMRPV      200     A
CCOMRPV            ZR     B6,RPV7     IF END OF TABLE                    002140   CCOMRPV      201     A
CCOMRPV            ZR     X2,RPV6     IF FILE DESELECTED BY CONTROL CARD OPTION 002150   CCOMRPV      202     A
CCOMRPV            SA1    X2          (X1) = FET WORD 1                  002160   CCOMRPV      203     A
CCOMRPV            MX0    -6                                             002170   CCOMRPV      204     A
CCOMRPV            SX6    B1                                             002180   CCOMRPV      205     A
CCOMRPV            BX7    X1+X6                                          002190   CCOMRPV      206     A
CCOMRPV            SA3    RPVA        (X3) = SHIFT TEST MASK FOR CIO CODE 002200   CCOMRPV      207     A
CCOMRPV            AX1    2                                              002210   CCOMRPV      208     A
CCOMRPV            BX0    -X0*X1      EXTRACT CIO CODE                   002220   CCOMRPV      209     A
CCOMRPV            SB2    X0                                             002230   CCOMRPV      210     A
CCOMRPV            LX3    B2                                             002240   CCOMRPV      211     A
CCOMRPV            PL     X3,RPV6     IF LAST CIO OP NOT OPEN OR WRITE   002250   CCOMRPV      212     A
CCOMRPV            SA7    A1          INSURE CIO COMPLETE BIT IS ON      002260   CCOMRPV      213     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCOMRPV

```
CCOMRPV            WRITER A1,,RCL     FLUSH BUFFER                          002270   CCOMRPV      214     A
CCOMRPV   RPV6     SB7    B7+1                                              002280   CCOMRPV      215     A
CCOMRPV            EQ     RPV5        CHECK NEXT ENTRY                      002290   CCOMRPV      216     A
CCOMRPV   RPV7     BSS    0                                                 002300   CCOMRPV      217     A
CCOMRPV   #RM      ENDIF                                                    002310   CCOMRPV      218     A
CCOMRPV   #RPV=    ENDIF                                                    002320   CCOMRPV      219     A
CCOMRPV                                                                     002330   CCOMRPV      220     A
CCOMRPV   *        RESTORE SAVED ENTRY POINT CONTENTS.                      002340   CCOMRPV      221     A
CCOMRPV                                                                     002350   CCOMRPV      222     A
CCOMRPV            SA1    RPVE                                              002360   CCOMRPV      223     A
CCOMRPV            SA2    A1+1                                              002370   CCOMRPV      224     A
CCOMRPV            BX6    X1                                                002380   CCOMRPV      225     A
CCOMRPV            LX7    X2                                                002390   CCOMRPV      226     A
CCOMRPV            SA6    =XCDD                                             002400   CCOMRPV      227     A
CCOMRPV            SA7    =XCOD                                             002410   CCOMRPV      228     A
CCOMRPV                                                                     002420   CCOMRPV      229     A
CCOMRPV   #OS      IFNE   .OS,2       IF NOT SCOPE 2                       002430   CCOMRPV      230     A
CCOMRPV            SA1    A2+B1       SYS=                                 002440   CCOMRPV      231     A
CCOMRPV            SA2    A1+B1       WNB=                                 002450   CCOMRPV      232     A
CCOMRPV            BX6    X1                                               002460   CCOMRPV      233     A
CCOMRPV            LX7    X2                                               002470   CCOMRPV      234     A
CCOMRPV            SA1    A2+1        MSG=                                 002480   CCOMRPV      235     A
CCOMRPV            SA6    =XSYS=                                           002490   CCOMRPV      236     A
CCOMRPV            SA7    =XWNB=                                           002500   CCOMRPV      237     A
CCOMRPV            BX7    X1                                               002510   CCOMRPV      238     A
CCOMRPV            SA7    =XMSG=                                           002520   CCOMRPV      239     A
CCOMRPV                                                                     002530   CCOMRPV      240     A
CCOMRPV   #RM      IFEQ   CP#RM,0     IF DIRECT CIO I/O                    002540   CCOMRPV      241     A
CCOMRPV            SA1    A2+B1                                            002550   CCOMRPV      242           I
          -CCG37
CCOMRPV            SA1    A1+B1                                            002620   CCG37          1     A
CCOMRPV            BX6    X1                                               002560   CCOMRPV      243     A
CCOMRPV            SA6    =XCIO=                                           002570   CCOMRPV      244     A
CCOMRPV   #RM      ENDIF                                                    002580   CCOMRPV      245     A
CCOMRPV   #OS      ENDIF                                                    002590   CCOMRPV      246     A
CCOMRPV                                                                     002600   CCOMRPV      247     A
CCOMRPV   *        REINSTATE THE ERROR CONDITON.                            002610   CCOMRPV      248     A
CCOMRPV                                                                     002620   CCOMRPV      249     A
CCOMRPV   #OS      IFNE   .OS,2       IF NOT SCOPE 2                       002630   CCOMRPV      250     A
CCOMRPV            SA1    RPVF        RPV RESET WORD                       002640   CCOMRPV      251     A
CCOMRPV            BX6    X1                                               002650   CCOMRPV      252     A
CCOMRPV   +        SA1    RA.MTR                                           002660   CCOMRPV      253     A
CCOMRPV            NZ     X1,*        WAIT FOR ALL CLEAR                   002670   CCOMRPV      254     A
CCOMRPV            SA6    A1          REQUEST RPV RESET                    002680   CCOMRPV      255     A
CCOMRPV   +        EQ     *           WAIT FOR OP SYS TO PICK UP ERROR     002690   CCOMRPV      256     A
CCOMRPV                                                                     002700   CCOMRPV      257     A
CCOMRPV   #OS      ELSE                                                     002710   CCOMRPV      258     A
CCOMRPV            SA1    XJP         EXCHANGE JUMP PACKAGE                002720   CCOMRPV      259     A
CCOMRPV            RJ     =XDXP=      DUMP EXCHANGE PACKAGE                002730   CCOMRPV      260     A
CCOMRPV            ABORT              *** TEMPORARY UNTIL SCOPE 2.0 REPRIEVE 002740   CCOMRPV      261     A
CCOMRPV   *                           *** RESET METHOD IS AVAILABLE.       002750   CCOMRPV      262     A
CCOMRPV   #OS      ENDIF                                                    002760   CCOMRPV      263     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCOMRPV

```
CCOMRPV                                                                  002770   CCOMRPV      264     A
CCOMRPV   #OS      IFNE   .OS,2       IF NOT SCOPE 2                    002780   CCOMRPV      265     A
CCOMRPV   RPVA     BSS    0                                              002790   CCOMRPV      266     A
CCOMRPV            ECHO   2,CIOCODE=(4B,14B,104B,120B,144B,160B)         002800   CCOMRPV      267     A
CCOMRPV            POS    60-CIOCODE_S-2                                 002810   CCOMRPV      268     A
CCOMRPV            VFD    1/1                                            002820   CCOMRPV      269     A
CCOMRPV            POS    0                                              002830   CCOMRPV      270     A
CCOMRPV            BSS    0                                              002840   CCOMRPV      271     A
CCOMRPV   #OS      ENDIF                                                 002850   CCOMRPV      272     A
CCOMRPV                                                                  002860   CCOMRPV      273     A
CCOMRPV            IF     DEF,RPV=CLN,1                                  002870   CCOMRPV      274     A
CCOMRPV   RPVB     DIS    ,* LAST STATEMENT BEGAN AT LINE ........*      002880   CCOMRPV      275     A
CCOMRPV                                                                  002890   CCOMRPV      276     A
CCOMRPV   RPVC     DIS    ,* ERROR AT 000000 IN XXXXXXX*                 002900   CCOMRPV      277     A
CCOMRPV                                                                  002910   CCOMRPV      278     A
CCOMRPV            IF     DEF,RPV=LOL,1                                  002920   CCOMRPV      279     A
CCOMRPV   RPVD     DIS    ,* LAST OVERLAY LOADED - (0,0)*                002930   CCOMRPV      280     A
CCOMRPV                                                                  002940   CCOMRPV      281     A
CCOMRPV   RPVE     BSSZ   2           TO SAVE (CDD) AND (COD)           002950   CCOMRPV      282     A
CCOMRPV                                                                  002960   CCOMRPV      283     A
CCOMRPV   #OS      IFNE   .OS,2       IF NOT SCOPE 2                    002970   CCOMRPV      284     A
CCOMRPV            BSSZ   3           TO SAVE (SYS=), (WNB=) AND (MSG=) 002980   CCOMRPV      285     A
CCOMRPV                                                                  002990   CCOMRPV      286     A
CCOMRPV   #RM      IFEQ   CP#RM,0     IF DIRECT CIO I/O                 003000   CCOMRPV      287     A
CCOMRPV            BSSZ   1           TO SAVE (CIO=)                    003010   CCOMRPV      288     A
CCOMRPV   #RM      ENDIF                                                 003020   CCOMRPV      289     A
CCOMRPV   RPVF     VFD    18/3RRPV,6/0,1/1,35/0    RPV RESET REQUEST WORD 003030   CCOMRPV      290     A
CCOMRPV   #OS      ENDIF                                                 003040   CCOMRPV      291     A
CCOMRPV   FRA      SPACE  4,8                                            003050   CCOMRPV      292     A
CCOMRPV  **        FRA - FIND RELATIVE ADDRESS.                          003060   CCOMRPV      293     A
CCOMRPV  *                                                               003070   CCOMRPV      294     A
CCOMRPV  *                GIVEN AN ABSOLUTE ADDRESS, *FRA=* FINDS THE ROUTINE 003080   CCOMRPV      295     A
CCOMRPV  *         NAME AND ABSOLUTE ADDRESS BY SEARCHING THE TABLES INSTALLED 003090   CCOMRPV      296     A
CCOMRPV  *         ORIGINALLY FOR *RPV=* PROCESSING.                     003100   CCOMRPV      297     A
CCOMRPV  *                                                               003110   CCOMRPV      298     A
CCOMRPV  *                *FRA* DOES NOT USE A RETURN JUMP CALLING SEQUENCE 003120   CCOMRPV      299     A
CCOMRPV  *         BECAUSE IT IS LOCATED IN CODE SPACE CHECKSUMMED BY *RPV=*. 003130   CCOMRPV      300     A
CCOMRPV  *                                                               003140   CCOMRPV      301     A
CCOMRPV  *                                                               003150   CCOMRPV      302     A
CCOMRPV  *         ENTRY  (X1) = ADDRESS RELATIVE TO RA+0                003160   CCOMRPV      303     A
CCOMRPV  *                (B7) = EXIT ADDRESS                            003170   CCOMRPV      304     A
CCOMRPV  *                                                               003180   CCOMRPV      305     A
CCOMRPV  *         EXIT   TO (B7), WITH ...                              003190   CCOMRPV      306     A
CCOMRPV  *                (A1,X1) = ADDRESS,CONTENTS OF RPV=RNA ENTRY    003200   CCOMRPV      307     A
CCOMRPV  *                (X4) = ABSOLUTE ADDRESS, H FORMAT              003210   CCOMRPV      308     A
CCOMRPV  *                (X6) = ABSOLUTE ADDRESS, DPC, *NNNNNN IN *     003220   CCOMRPV      309     A
CCOMRPV  *                (X7) = ROUTINE NAME, DPC, L FORMAT             003230   CCOMRPV      310     A
CCOMRPV  *                                                               003240   CCOMRPV      311     A
CCOMRPV  *         USES   X - 1, 2, 3, 4, 6, 7                           003250   CCOMRPV      312     A
CCOMRPV  *                A - 2, 4                                       003260   CCOMRPV      313     A
CCOMRPV  *                B - 2, 3, 4, 5                                 003270   CCOMRPV      314     A
CCOMRPV  *                                                               003280   CCOMRPV      315     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCOMRPV

```
CCOMRPV  *         CANNOT DESTROY  B1  X0,5  A0,5,6,7                    003290  CCOMRPV  316  A
CCOMRPV  *                                                               003300  CCOMRPV  317  A
CCOMRPV  *         CALLS  COD                                            003310  CCOMRPV  318  A
CCOMRPV                                                                  003320  CCOMRPV  319  A
CCOMRPV                                                                  003330  CCOMRPV  320  A
CCOMRPV   #RPV=    IF     DEF,RPV=RNA IF ROUTINE NAME ADDRESS TABLE EXISTS 003340  CCOMRPV  321  A
CCOMRPV   FRA=     BSS    0           ...ENTRY                          003350  CCOMRPV  322  A
CCOMRPV            SA2    RPV=RNA     (X2) = FWA OF 1ST RNA TABLE       003360  CCOMRPV  323  A
CCOMRPV            SB2    X1          (B2) = ADDRESS RELATIVE TO RA+0   003370  CCOMRPV  324  A
CCOMRPV            SA2    X2          (A2,X2) = A+C OF 1ST ENTRY IN RNA TABLE 003380  CCOMRPV  325  A
CCOMRPV            SB3    X2                                            003390  CCOMRPV  326  A
CCOMRPV            SB4    B0                                            003400  CCOMRPV  327  A
CCOMRPV            SB5    A2+                                           003410  CCOMRPV  328  A
CCOMRPV                                                                  003420  CCOMRPV  329  A
CCOMRPV  *         SEARCH ROUTINE NAME ADDRESS TABLE.                   003430  CCOMRPV  330  A
CCOMRPV                                                                  003440  CCOMRPV  331  A
CCOMRPV   FRA2     BSS                                                   003450  CCOMRPV  332  A
CCOMRPV            ZR     X2,FRA3     IF END OF BLOCK                   003460  CCOMRPV  333  A
CCOMRPV            SB3    X2                                            003470  CCOMRPV  334  A
CCOMRPV            SA2    A2+B1       (X2) = NEXT TABLE ENTRY           003480  CCOMRPV  335  A
CCOMRPV            GT     B3,B2,FRA2  IF CURRENT ENTRY BEYOND ABS ADDR  003490  CCOMRPV  336  A
CCOMRPV            GE     B4,B3,FRA2  IF CURRENT NOT CLOSER             003500  CCOMRPV  337  A
CCOMRPV            SB5    A2-B1       CLOSEST SO FAR                    003510  CCOMRPV  338  A
CCOMRPV            SB4    B3                                            003520  CCOMRPV  339  A
CCOMRPV            EQ     FRA2        KEEP LOOKING                      003530  CCOMRPV  340  A
CCOMRPV                                                                  003540  CCOMRPV  341  A
CCOMRPV  *         CHECK FOR TABLE CONTINUATION LINK.                   003550  CCOMRPV  342  A
CCOMRPV                                                                  003560  CCOMRPV  343  A
CCOMRPV   FRA3     BSS                                                   003570  CCOMRPV  344  A
CCOMRPV            SA2    A2+B1       END OF TABLE/TABLE LINK WORD      003580  CCOMRPV  345  A
CCOMRPV            ZR     X2,FRA4     IF END OF TABLE                   003590  CCOMRPV  346  A
CCOMRPV            SA2    X2          (A2,X2) = A+C OF 1ST WORD IN NEXT BLOCK 003600  CCOMRPV  347  A
CCOMRPV            EQ     FRA2        CONTINUE TABLE SEARCH...          003610  CCOMRPV  348  A
CCOMRPV                                                                  003620  CCOMRPV  349  A
CCOMRPV  *         EXTRACT ROUTINE NAME AND FORMAT ABSOLUTE ADDRESS.    003630  CCOMRPV  350  A
CCOMRPV                                                                  003640  CCOMRPV  351  A
CCOMRPV   FRA4     BSS                                                   003650  CCOMRPV  352  A
CCOMRPV            SX1    B2-B4       (X1) = BIN ADDR, RELATIVE TO CLOSEST FIND 003660  CCOMRPV  353  A
CCOMRPV            RJ     =XCOD       CONVERT TO OCTAL DISPLAY CODE     003670  CCOMRPV  354  A
CCOMRPV            SA1    B5          (X1) = 42/0LNAME,18/ADDRESS       003680  CCOMRPV  355  A
CCOMRPV            MX7    42                                            003690  CCOMRPV  356  A
CCOMRPV            SX3    2R  &2RIN                                     003700  CCOMRPV  357  A
CCOMRPV            BX7    X7*X1       (X7) = ROUTINE NAME, 0L FORMAT    003710  CCOMRPV  358  A
CCOMRPV            LX3    6                                             003720  CCOMRPV  359  A
CCOMRPV            BX6    X4-X3       (X6) = RELATIVE ADDRESS, *NNNNNN IN * 003730  CCOMRPV  360  A
CCOMRPV            JP     B7          EXIT ...                          003740  CCOMRPV  361  A
CCOMRPV   #RPV=    ENDIF                                                 003750  CCOMRPV  362  A
CCOMRPV   RPV      SPACE  4,8                                           003760  CCOMRPV  363  A
CCOMRPV   RPV=     SPACE  4,10                                          003770  CCOMRPV  364  A
CCOMRPV            BSS    0                                             003780  CCOMRPV  365  A
CCOMRPV   LWARPV   =      *-1         *** END OF CHECKSUMMED REPRIEVE CODE *** 003790  CCOMRPV  366  A
CCOMRPV   RPV=     SPACE  4,10                                          003800  CCOMRPV  367  A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCOMRPV

CCOMRPV   QUAL$    IF     -DEF,QUAL$                                    003810   CCOMRPV      368     A
CCOMRPV            QUAL   *                                             003820   CCOMRPV      369     A
CCOMRPV   RPV=     EQU    /CCOMRPV/RPV=                                 003830   CCOMRPV      370     A
CCOMRPV   FRA=     EQU    /CCOMRPV/FRA=                                 003840   CCOMRPV      371     A
CCOMRPV   QUAL$    ENDIF                                                003850   CCOMRPV      372     A
CCOMRPV   CCOMRPV  ENDX                                                 003860   CCOMRPV      373     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - CCOMRPV

          IDENTIFIER                TOTAL    ACTIVE

          CCOMRPV                     373       372
          CCG37                         1         1

          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=MAC

DBG=MAC  *COMDECK DBG=MAC                                                                DBG=MAC        1     A
DBG=MAC   DBG=MAC  CTEXT  DBG=MAC - INTERACTIVE AND BATCH DEBUG PACKAGE MACROS.          DBG=MAC        2     A
DBG=MAC   DBG=MAC  SPACE  4,8                                                            DBG=MAC        3     A
DBG=MAC            IF     -DEF,QUAL$,1                                                   DBG=MAC        4     A
DBG=MAC            QUAL   DBG=MAC                                                        DBG=MAC        5     A
DBG=MAC            BASE   D                                                              DBG=MAC        6     A
DBG=MAC   DBG=MAC  SPACE  4,8                                                            DBG=MAC        7     A
DBG=MAC  **        DBG=MAC - INTERACTIVE AND BATCH DEBUG PACKAGE MACROS.                 DBG=MAC        8     A
DBG=MAC  *                                                                               DBG=MAC        9     A
DBG=MAC  *                                                                               DBG=MAC       10     A
DBG=MAC  *                CONTAINS THE MACROS USED TO ACCESS THE ROUTINES IN             DBG=MAC       11     A
DBG=MAC  *         *DBG=IDP*--                                                           DBG=MAC       12     A
DBG=MAC  *                                                                               DBG=MAC       13     A
DBG=MAC  * NAME    BREAK  LL,UL,INC   TO PLACE A BREAKPOINT                              DBG=MAC       14     A
DBG=MAC  * NAME    REG                TO PROVIDE A SNAPSHOT OF ALL REGISTERS             DBG=MAC       15     A
DBG=MAC  * NAME    REG    (X1,X2,A4)  TO PROVIDE A SNAPSHOT OF SELECTED REGISTERS         DBG=MAC       16     A
DBG=MAC  * NAME    SNAP   FWA,LWA,LEN TO PROVIDE A SNAPSHOT OF CENTRAL MEMORY             DBG=MAC       17     A
DBG=MAC   BREAK    SPACE  4,8                                                            DBG=MAC       18     A
DBG=MAC  **        BREAK - PLACE A BREAKPOINT.                                           DBG=MAC       19     A
DBG=MAC  *                                                                               DBG=MAC       20     A
DBG=MAC  *                                                                               DBG=MAC       21     A
DBG=MAC  *                THIS MACRO DOES A RETURN JUMP TO THE INTERACTIVE               DBG=MAC       22     A
DBG=MAC  *         DEBUG PACKAGE *IDP*.                                                  DBG=MAC       23     A
DBG=MAC  *                                                                               DBG=MAC       24     A
DBG=MAC  * NAME    BREAK  LL,UL,INC                                                      DBG=MAC       25     A
DBG=MAC  *                                                                               DBG=MAC       26     A
DBG=MAC  *         ENTRY  NAME = DISPLAY CODE (DPC) MESSAGE TO BE PRINTED WITH           DBG=MAC       27     A
DBG=MAC  *                       EACH SNAP. (NR CHARS.LE.8)                              DBG=MAC       28     A
DBG=MAC  *                LL   = LOWER LIMIT (DEFAULT=1)                                 DBG=MAC       29     A
DBG=MAC  *                UL   = UPPER LIMIT (DEFAULT=100)                               DBG=MAC       30     A
DBG=MAC  *                NUSE =  USE BLOCK FLAG. IF *NUSE* IS NULL, THE PARA-           DBG=MAC       31     A
DBG=MAC  *                         METER LIST EXISTS IN *USE DEBUG*, ELSE IS             DBG=MAC       32     A
DBG=MAC  *                         ASSEMBLED AFTER *RJ*                                  DBG=MAC       33     A
DBG=MAC  *                                                                               DBG=MAC       34     A
DBG=MAC  *         EXIT   NONE                                                           DBG=MAC       35     A
DBG=MAC  *                                                                               DBG=MAC       36     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=MAC

```
DBG=MAC  *         USES   NONE                                                  DBG=MAC    37   A
DBG=MAC  *                                                                      DBG=MAC    38   A
DBG=MAC  *         CALLS  IDP=                                                  DBG=MAC    39   A
DBG=MAC                                                                         DBG=MAC    40   A
DBG=MAC                                                                         DBG=MAC    41   A
DBG=MAC            PURGMAC BREAK                                                DBG=MAC    42   A
DBG=MAC                                                                         DBG=MAC    43   A
DBG=MAC            MACRO  BREAK,NAME,LL,UL,INC,NUSE                             DBG=MAC    44   A
DBG=MAC            LOCAL  RTN                                                   DBG=MAC    45   A
DBG=MAC            LOCAL  APL                                                   DBG=MAC    46   A
DBG=MAC  *                                                                      DBG=MAC    47   A
DBG=MAC   +        RJ     =XIDP=                                                DBG=MAC    48   A
DBG=MAC   -        VFD    30/APL                                                DBG=MAC    49   A
DBG=MAC  *                                                                      DBG=MAC    50   A
DBG=MAC   #NUSE    IFC    EQ,/NUSE//                                            DBG=MAC    51   A
DBG=MAC            USE    DEBUG                                                 DBG=MAC    52   A
DBG=MAC  *                                                                      DBG=MAC    53   A
DBG=MAC   #NUSE    ELSE                                                         DBG=MAC    54   A
DBG=MAC            EQ     RTN                                                   DBG=MAC    55   A
DBG=MAC   #NUSE    ENDIF                                                        DBG=MAC    56   A
DBG=MAC   APL      BSS    0                                                     DBG=MAC    57   A
DBG=MAC            FRK=   (NAME),LL,UL,INC                                      DBG=MAC    58   A
DBG=MAC  *                                                                      DBG=MAC    59   A
DBG=MAC   #NUSE    IFC    EQ,/NUSE//                                            DBG=MAC    60   A
DBG=MAC            USE    *                                                     DBG=MAC    61   A
DBG=MAC  *                                                                      DBG=MAC    62   A
DBG=MAC   #NUSE    ELSE                                                         DBG=MAC    63   A
DBG=MAC   RTN      BSS    0                                                     DBG=MAC    64   A
DBG=MAC   #NUSE    ENDIF                                                        DBG=MAC    65   A
DBG=MAC  *                                                                      DBG=MAC    66   A
DBG=MAC   BREAK    ENDM                                                         DBG=MAC    67   A
DBG=MAC   FRK=     SPACE  4,8                                                   DBG=MAC    68   A
DBG=MAC  **        FRK= - GENERATE FREQUENCY PARAMETER LIST.                    DBG=MAC    69   A
DBG=MAC  *                                                                      DBG=MAC    70   A
DBG=MAC  *                                                                      DBG=MAC    71   A
DBG=MAC  *         FRK=   (NAME),LL,UL,INC                                      DBG=MAC    72   A
DBG=MAC  *                                                                      DBG=MAC    73   A
DBG=MAC  *         ENTRY  NAME =  DISPLAY CODE (DPC) MESSAGE TO BE PRINTED WITH DBG=MAC    74   A
DBG=MAC  *                          EACH SNAP. (NR CHARS.LE.8)                  DBG=MAC    75   A
DBG=MAC  *                LL   =  LOWER LIMIT (DEFAULT=1)                       DBG=MAC    76   A
DBG=MAC  *                UL   =  UPPER LIMIT (DEFAULT=100)                     DBG=MAC    77   A
DBG=MAC  *                INC  =  INCREMENT (DEFAULT=1)                         DBG=MAC    78   A
DBG=MAC  *                          LL, UL, AND INC ARE THE SNAP FREQUENCY PAR-  DBG=MAC    79   A
DBG=MAC  *                          AMETERS AND ARE RELATED AS FOLLOWS--  THE 1ST DBG=MAC    80   A
DBG=MAC  *                          SNAP PRINTED WILL OCCUR THE LLTH TIME THIS   DBG=MAC    81   A
DBG=MAC  *                          SNAP IS EXECUTED. THE 2ND SNAP PRINTED WILL  DBG=MAC    82   A
DBG=MAC  *                          OCCUR THE (LL+INC)TH TIME THIS SNAP IS EXEC- DBG=MAC    83   A
DBG=MAC  *                          UTED, AND SO ON. THE LAST SNAP PRINTED WILL  DBG=MAC    84   A
DBG=MAC  *                          OCCUR THE ULTH TIME THIS SNAP IS EXECUTED.   DBG=MAC    85   A
DBG=MAC  *                                                                      DBG=MAC    86   A
DBG=MAC  *         EXIT   NONE                                                  DBG=MAC    87   A
DBG=MAC  *                                                                      DBG=MAC    88   A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=MAC

```
DBG=MAC  *         USES   NONE                                                    DBG=MAC      89    A
DBG=MAC  *                                                                        DBG=MAC      90    A
DBG=MAC  *         CALLS  NONE                                                    DBG=MAC      91    A
DBG=MAC                                                                           DBG=MAC      92    A
DBG=MAC                                                                           DBG=MAC      93    A
DBG=MAC            PURGMAC FRK=                                                   DBG=MAC      94    A
DBG=MAC                                                                           DBG=MAC      95    A
DBG=MAC   FRK=     MACRO  NAME,LL,UL,INC                                          DBG=MAC      96    A
DBG=MAC            CON    LL  1                                                   DBG=MAC      97    A
DBG=MAC            CON    UL  100                                                 DBG=MAC      98    A
DBG=MAC            CON    INC 1                                                   DBG=MAC      99    A
DBG=MAC   .1       IFC    EQ,/NAME//                                              DBG=MAC     100    A
DBG=MAC            CON    0                                                       DBG=MAC     101    A
DBG=MAC   .1       ELSE                                                           DBG=MAC     102    A
DBG=MAC            CON    10H_NAME                                                DBG=MAC     103    A
DBG=MAC   .1       ENDIF                                                          DBG=MAC     104    A
DBG=MAC            CON    0                                                       DBG=MAC     105    A
DBG=MAC  *                         USED BY *FRK=/TSNAP* TO KEEP SNAP COUNT        DBG=MAC     106    A
DBG=MAC   FRK=     ENDM                                                           DBG=MAC     107    A
DBG=MAC   REG      SPACE  4,8                                                     DBG=MAC     108    A
DBG=MAC  **        REG - REGISTER SNAPSHOT.                                       DBG=MAC     109    A
DBG=MAC  *                                                                        DBG=MAC     110    A
DBG=MAC  *                                                                        DBG=MAC     111    A
DBG=MAC  * NAME    REG    RG,LL,UL,INC,NUSE,UF                            000170  DBG=MAC     112    A
DBG=MAC  *                                                                        DBG=MAC     113    A
DBG=MAC  *         REG                                                            DBG=MAC     114    A
DBG=MAC  *         REG    (X1,B3,A4)                                              DBG=MAC     115    A
DBG=MAC  *         REG    NOREG                                                   DBG=MAC     116    A
DBG=MAC  *                                                                        DBG=MAC     117    A
DBG=MAC  *         ENTRY  NAME =  DISPLAY CODE (DPC) MESSAGE TO BE PRINTED WITH   DBG=MAC     118    A
DBG=MAC  *                          EACH SNAP. (NR CHARS.LE.8)                    DBG=MAC     119    A
DBG=MAC  *                    =  NULL IF NO SNAP HEADER                           DBG=MAC     120    A
DBG=MAC  *                RG  =  NULL IF ALL REGISTERS ARE TO BE DUMPED           DBG=MAC     121    A
DBG=MAC  *                    =  A LIST OF THE REGISTERS (ENCLOSED IN PARENS)     DBG=MAC     122    A
DBG=MAC  *                          DESIRED TO BE DUMPED                          DBG=MAC     123    A
DBG=MAC  *                    =  NON-REGISTER/NON-NULL IF NO REGISTERS TO BE      DBG=MAC     124    A
DBG=MAC  *                          DUMPED                                        DBG=MAC     125    A
DBG=MAC  *                LL  =  LOWER LIMIT (DEFAULT=1)                          DBG=MAC     126    A
DBG=MAC  *                UL  =  UPPER LIMIT (DEFAULT=100)                        DBG=MAC     127    A
DBG=MAC  *                NUSE =  USE BLOCK FLAG. IF *NUSE* IS NULL, THE PARA-    DBG=MAC     128    A
DBG=MAC  *                          METER LIST EXISTS IN *USE DEBUG*, ELSE IS     DBG=MAC     129    A
DBG=MAC  *                          ASSEMBLED AFTER *RJ*                          DBG=MAC     130    A
DBG=MAC  *                UF  =  A STRING OF USER SNAP FLAGS PROCESSED BY 000190  DBG=MAC     131    A
DBG=MAC  *                       INSTALLER SUPPLIED USF= MACRO IF DESIRED.  000200  DBG=MAC     132    A
DBG=MAC  *                       USF= SHOULD GENERATE 1 WORD.               000210  DBG=MAC     133    A
DBG=MAC  *                                                                        DBG=MAC     134    A
DBG=MAC  *         EXIT   NONE                                                    DBG=MAC     135    A
DBG=MAC  *                                                                        DBG=MAC     136    A
DBG=MAC  *         USES   NONE                                                    DBG=MAC     137    A
DBG=MAC  *                                                                        DBG=MAC     138    A
DBG=MAC  *         CALLS  FRK=,REG=,RGR=,USF=(IF DEF)                      000212  DBG=MAC     139    A
DBG=MAC                                                                           DBG=MAC     140    A
```

```
         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=MAC

```
DBG=MAC                                                                                DBG=MAC      141     A
DBG=MAC            PURGMAC REG                                                         DBG=MAC      142     A
DBG=MAC                                                                                DBG=MAC      143     A
DBG=MAC            MACRO  REG,NAME,RG,LL,UL,INC,NUSE,UF                       000214   DBG=MAC      144     A
DBG=MAC            LOCAL  '?ID#001,'?ID#004                                   000215   DBG=MAC      145     A
DBG=MAC            LOCAL  RTN                                                          DBG=MAC      146     A
DBG=MAC            LOCAL  APL                                                          DBG=MAC      147     A
DBG=MAC  *                                                                             DBG=MAC      148     A
DBG=MAC   +        RJ     =XREG=                                                       DBG=MAC      149     A
DBG=MAC   -        VFD    30/APL                                                       DBG=MAC      150     A
DBG=MAC  *                                                                             DBG=MAC      151     A
DBG=MAC   #NUSE    IFC    EQ,/NUSE//                                                   DBG=MAC      152     A
DBG=MAC            USE    DEBUG                                                        DBG=MAC      153     A
DBG=MAC  *                                                                             DBG=MAC      154     A
DBG=MAC   #NUSE    ELSE                                                                DBG=MAC      155     A
DBG=MAC            EQ     RTN                                                          DBG=MAC      156     A
DBG=MAC   #NUSE    ENDIF                                                               DBG=MAC      157     A
DBG=MAC   APL      BSS    0                                                            DBG=MAC      158     A
DBG=MAC            FRK=   (NAME),LL,UL,INC                                             DBG=MAC      159     A
DBG=MAC   '?ID#001 BSS 0                                                      000230   DBG=MAC      160     A
DBG=MAC            USF=   (UF)                                                 000240   DBG=MAC      161     A
DBG=MAC  *                                                                    000241   DBG=MAC      162     A
DBG=MAC   '?ID#002 IFNE *P,60D                                                000242   DBG=MAC      163     A
DBG=MAC            IFNE *P,0,2                                                000243   DBG=MAC      164     A
DBG=MAC            POS 0                                                      000244   DBG=MAC      165     A
DBG=MAC   '?ID#003 SKIP                                                       000245   DBG=MAC      166     A
DBG=MAC  *                                                                    000246   DBG=MAC      167     A
DBG=MAC   '?ID#002 ENDIF                                                      000247   DBG=MAC      168     A
DBG=MAC   '?ID#004 BSS 0                                                      000248   DBG=MAC      169     A
DBG=MAC  *                                                                    000249   DBG=MAC      170     A
DBG=MAC            IFNE   '?ID#004-'?ID#001,1,2                               000250   DBG=MAC      171     A
DBG=MAC   '?ID#003 ENDIF                                                      000251   DBG=MAC      172     A
DBG=MAC   P        ERR    *USF=* DID NOT GENERATE EXACTLY 60D BITS. "SEQUENCE"  000252   DBG=MAC      173     A
DBG=MAC  *                                                                    000253   DBG=MAC      174     A
DBG=MAC            RGR=   (RG)                                                         DBG=MAC      175     A
DBG=MAC  *                                                                             DBG=MAC      176     A
DBG=MAC   #NUSE    IFC    EQ,/NUSE//                                                   DBG=MAC      177     A
DBG=MAC            USE    *                                                            DBG=MAC      178     A
DBG=MAC  *                                                                             DBG=MAC      179     A
DBG=MAC   #NUSE    ELSE                                                                DBG=MAC      180     A
DBG=MAC   RTN      BSS    0                                                            DBG=MAC      181     A
DBG=MAC   #NUSE    ENDIF                                                               DBG=MAC      182     A
DBG=MAC  *                                                                             DBG=MAC      183     A
DBG=MAC   REG      ENDM                                                                DBG=MAC      184     A
DBG=MAC                                                                                DBG=MAC      185     A
DBG=MAC            PURGMAC REGS                                                        DBG=MAC      186     A
DBG=MAC   REGS     OPSYN  REG                                                          DBG=MAC      187     A
DBG=MAC   RGR=     SPACE  4,8                                                          DBG=MAC      188     A
DBG=MAC  **        RGR= - GENERATE REGISTER PARAMETER LIST.                            DBG=MAC      189     A
DBG=MAC  *                                                                             DBG=MAC      190     A
DBG=MAC  *                                                                             DBG=MAC      191     A
DBG=MAC  *                SEE *DSR/DBG=IDP*.                                           DBG=MAC      192     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=MAC

```
DBG=MAC  *                                                                  DBG=MAC    193    A
DBG=MAC  *          RGR=   RG                                               DBG=MAC    194    A
DBG=MAC  *                                                                  DBG=MAC    195    A
DBG=MAC  *          RGR=                                                    DBG=MAC    196    A
DBG=MAC  *          RGR=   (X1,X2,A3,B4)                                    DBG=MAC    197    A
DBG=MAC  *          RGR=   NOREG                                            DBG=MAC    198    A
DBG=MAC  *                                                                  DBG=MAC    199    A
DBG=MAC  *          ENTRY  RG   =  NULL IF ALL REGISTERS ARE TO BE DUMPED   DBG=MAC    200    A
DBG=MAC  *                     =  A LIST OF THE REGISTERS (ENCLOSED IN PARENS) DBG=MAC  201    A
DBG=MAC  *                        DESIRED TO BE DUMPED                      DBG=MAC    202    A
DBG=MAC  *                     =  NON-REGISTER/NON-NULL IF NO REGISTERS TO BE DBG=MAC  203    A
DBG=MAC  *                        DUMPED                                    DBG=MAC    204    A
DBG=MAC  *                                                                  DBG=MAC    205    A
DBG=MAC  *          EXIT   NONE                                             DBG=MAC    206    A
DBG=MAC  *                                                                  DBG=MAC    207    A
DBG=MAC  *          USES   NONE                                             DBG=MAC    208    A
DBG=MAC  *                                                                  DBG=MAC    209    A
DBG=MAC  *          CALLS  NONE                                             DBG=MAC    210    A
DBG=MAC                                                                     DBG=MAC    211    A
DBG=MAC                                                                     DBG=MAC    212    A
DBG=MAC             PURGMAC RGR=                                            DBG=MAC    213    A
DBG=MAC                                                                     DBG=MAC    214    A
DBG=MAC   RGR=      MACRO  RG                                               DBG=MAC    215    A
DBG=MAC  *                                                                  DBG=MAC    216    A
DBG=MAC   .1        IFC    EQ,/RG//                                         DBG=MAC    217    A
DBG=MAC             DATA   -1                                               DBG=MAC    218    A
DBG=MAC   .1        ELSE                                                    DBG=MAC    219    A
DBG=MAC  *                                                                  DBG=MAC    220    A
DBG=MAC   .2        IRP    RG                                               DBG=MAC    221    A
DBG=MAC   .3        IF     -REG,RG                                          DBG=MAC    222    A
DBG=MAC             DATA   0                                                DBG=MAC    223    A
DBG=MAC   .3        ELSE                                                    DBG=MAC    224    A
DBG=MAC   .T        MICRO  1,1,/RG/                                         DBG=MAC    225    A
DBG=MAC   .R        MICRO  2,1,/RG/                                         DBG=MAC    226    A
DBG=MAC             VFD    60/=XSV_".T"+".R"                                DBG=MAC    227         I
          -CCGA017
DBG=MAC             VFD    60/=XIDP=SV_".T"+".R"                            CCGA017      6    A
DBG=MAC   .3        ENDIF                                                   DBG=MAC    228    A
DBG=MAC   .2        IRP                                                     DBG=MAC    229    A
DBG=MAC  *                                                                  DBG=MAC    230    A
DBG=MAC             DATA   0                                                DBG=MAC    231    A
DBG=MAC   .1        ENDIF                                                   DBG=MAC    232    A
DBG=MAC  *                                                                  DBG=MAC    233    A
DBG=MAC   RGR=      ENDM                                                    DBG=MAC    234    A
DBG=MAC   SNAP      SPACE  4,8                                              DBG=MAC    235    A
DBG=MAC  **         SNAP - SNAPSHOT OF REGISTERS AND CORE.                  DBG=MAC    236    A
DBG=MAC  *                                                                  DBG=MAC    237    A
DBG=MAC  *                                                                  DBG=MAC    238    A
DBG=MAC  * NAME     SNAP   FWA,LWA,LNG,RG,LL,UL,INC,NUSE,UF          000280  DBG=MAC    239    A
DBG=MAC  *          SNAP   *FWA,,**LNG,(X1,X2,B5,A1)                        DBG=MAC    240    A
DBG=MAC  *                                                                  DBG=MAC    241    A
DBG=MAC  *          ENTRY  NAME =  DISPLAY CODE (DPC) MESSAGE TO BE PRINTED WITH DBG=MAC  242    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=MAC

```
DBG=MAC  *                        EACH SNAP. (NR CHARS.LE.8)                          DBG=MAC    243    A
DBG=MAC  *                    =  NULL IF NO SNAP HEADER                              DBG=MAC    244    A
DBG=MAC  *                FWA =  FWA OF CORE TO BE DUMPED. THIS PARAMETER MUST       DBG=MAC    245    A
DBG=MAC  *                        APPEAR IF A CORE DUMP IS DESIRED.                  DBG=MAC    246    A
DBG=MAC  *                LWA =  LWA OF CORE TO BE DUMPED. (OPTIONAL)                DBG=MAC    247    A
DBG=MAC  *                LNG =  NR OF WORDS TO BE DUMPED. *LWA* AND *LNG* ARE       DBG=MAC    248    A
DBG=MAC  *                        RELATED AS FOLLOWS--  DUMP IS FROM                 DBG=MAC    249    A
DBG=MAC  *                        FWA TO FWA+LNG UNLESS LNG IS NULL OR ZERO.         DBG=MAC    250    A
DBG=MAC  *                        IN THAT CASE, DUMP IS FROM FWA TO LWA.             DBG=MAC    251    A
DBG=MAC  *                RG  =  NULL IF ALL REGISTERS ARE TO BE DUMPED              DBG=MAC    252    A
DBG=MAC  *                    =  A LIST OF THE REGISTERS (ENCLOSED IN PARENS)        DBG=MAC    253    A
DBG=MAC  *                        DESIRED TO BE DUMPED                               DBG=MAC    254    A
DBG=MAC  *                    =  NON-REGISTER/NON-NULL IF NO REGISTERS TO BE         DBG=MAC    255    A
DBG=MAC  *                        DUMPED                                             DBG=MAC    256    A
DBG=MAC  *                LL  =  LOWER LIMIT (DEFAULT=1)                             DBG=MAC    257    A
DBG=MAC  *                UL  =  UPPER LIMIT (DEFAULT=100)                           DBG=MAC    258    A
DBG=MAC  *                INC =  INCREMENT (DEFAULT=1)                               DBG=MAC    259    A
DBG=MAC  *                NUSE =  USE BLOCK FLAG. IF *NUSE* IS NULL, THE PARA-       DBG=MAC    260    A
DBG=MAC  *                        METER LIST EXISTS IN *USE DEBUG*, ELSE IS          DBG=MAC    261    A
DBG=MAC  *                        ASSEMBLED AFTER *RJ*                               DBG=MAC    262    A
DBG=MAC  *                UF  =  A STRING OF USER SNAP FLAGS PROCESSED BY    000300 DBG=MAC    263    A
DBG=MAC  *                       INSTALLER SUPPLIED USF= MACRO IF DESIRED.   000310 DBG=MAC    264    A
DBG=MAC  *                       USF= SHOULD GENERATE 1 WORD.                000320 DBG=MAC    265    A
DBG=MAC  *                                                                          DBG=MAC    266    A
DBG=MAC  *         NOTES-- AN INDIRECT ADDRESSING SYNTAX IS PROVIDED FOR THE FWA,   DBG=MAC    267    A
DBG=MAC  *         LWA, AND LNG PARAMETERS. ANY OF THESE PARAMETERS PRECEDED BY     DBG=MAC    268    A
DBG=MAC  *         AN ASTERICK (*) DENOTES ONE LEVEL OF INDIRECT ADDRESSING.        DBG=MAC    269    A
DBG=MAC  *         N LEVELS OF INDIRECT ADDRESSING ARE POSSIBLE (N.LE.2**21-1)      DBG=MAC    270    A
DBG=MAC  *         BY STACKING ASTERICKS. (E.G.  SNAP  **X1,***LWA)                 DBG=MAC    271    A
DBG=MAC  *                                                                          DBG=MAC    272    A
DBG=MAC  *                A SINGLE REGISTER OR AN ADDRESS EXPRESSION CAN BE USED    DBG=MAC    273    A
DBG=MAC  *         FOR THE FWA, LWA, OR LNG PARAMETER, BUT A MIX (E.G. SNAP X1+1)   DBG=MAC    274    A
DBG=MAC  *         IS NOT ALLOWED.                                                  DBG=MAC    275    A
DBG=MAC  *                                                                          DBG=MAC    276    A
DBG=MAC  *         EXIT   NONE                                                      DBG=MAC    277    A
DBG=MAC  *                                                                          DBG=MAC    278    A
DBG=MAC  *         USES   NONE        (ALL REGISTERS ARE SAVED AND RESTORED)        DBG=MAC    279    A
DBG=MAC  *                                                                          DBG=MAC    280    A
DBG=MAC  *         CALLS  FRK=,RGR=,SNG=,SNP=,USF=(IF DEF)                    000323 DBG=MAC    281    A
DBG=MAC                                                                             DBG=MAC    282    A
DBG=MAC                                                                             DBG=MAC    283    A
DBG=MAC            PURGMAC SNAP                                                     DBG=MAC    284    A
DBG=MAC                                                                             DBG=MAC    285    A
DBG=MAC            MACRO  SNAP,NAME,FWA,LWA,LNG,RG,LL,UL,INC,NUSE,UF          000325 DBG=MAC    286    A
DBG=MAC            LOCAL  '?ID#005,'?ID#008                                   000326 DBG=MAC    287    A
DBG=MAC            LOCAL  RTN                                                       DBG=MAC    288    A
DBG=MAC            LOCAL  APL                                                       DBG=MAC    289    A
DBG=MAC  *                                                                          DBG=MAC    290    A
DBG=MAC   +        RJ     =XSNP=                                                    DBG=MAC    291    A
DBG=MAC   -        VFD    30/APL                                                    DBG=MAC    292    A
DBG=MAC  *                                                                          DBG=MAC    293    A
DBG=MAC   #NUSE    IFC    EQ,/NUSE//                                                DBG=MAC    294    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=MAC

```
DBG=MAC            USE    DEBUG                                                    DBG=MAC      295     A
DBG=MAC  *                                                                         DBG=MAC      296     A
DBG=MAC   #NUSE    ELSE                                                             DBG=MAC      297     A
DBG=MAC            EQ     RTN                                                      DBG=MAC      298     A
DBG=MAC   #NUSE    ENDIF                                                            DBG=MAC      299     A
DBG=MAC   APL      BSS    0                                                         DBG=MAC      300     A
DBG=MAC            FRK=   (NAME),LL,UL,INC                                         DBG=MAC      301     A
DBG=MAC            SNG=   (FWA)                                                    DBG=MAC      302     A
DBG=MAC            SNG=   (LWA)                                                    DBG=MAC      303     A
DBG=MAC            SNG=   (LNG)                                                    DBG=MAC      304     A
DBG=MAC   '?ID#005 BSS 0                                                  000340   DBG=MAC      305     A
DBG=MAC            USF=   (UF)                                            000350   DBG=MAC      306     A
DBG=MAC  *                                                                000352   DBG=MAC      307     A
DBG=MAC   '?ID#006 IFNE *P,60D                                            000353   DBG=MAC      308     A
DBG=MAC            IFNE *P,0,2                                            000354   DBG=MAC      309     A
DBG=MAC            POS 0                                                  000355   DBG=MAC      310     A
DBG=MAC   '?ID#007 SKIP                                                   000356   DBG=MAC      311     A
DBG=MAC  *                                                                000357   DBG=MAC      312     A
DBG=MAC   '?ID#006 ENDIF                                                  000358   DBG=MAC      313     A
DBG=MAC   '?ID#008 BSS 0                                                  000359   DBG=MAC      314     A
DBG=MAC  *                                                                000360   DBG=MAC      315     A
DBG=MAC            IFNE   '?ID#008-'?ID#005,1,2                           000361   DBG=MAC      316     A
DBG=MAC   '?ID#007 ENDIF                                                  000362   DBG=MAC      317     A
DBG=MAC   P        ERR    *USF= DID NOT GENERATE EXACTLY 60D BITS. "SEQUENCE"    000363   DBG=MAC      318     A
DBG=MAC  *                                                                000364   DBG=MAC      319     A
DBG=MAC            RGR=   (RG)                                                     DBG=MAC      320     A
DBG=MAC  *                                                                         DBG=MAC      321     A
DBG=MAC   #NUSE    IFC    EQ,/NUSE//                                               DBG=MAC      322     A
DBG=MAC            USE    *                                                        DBG=MAC      323     A
DBG=MAC  *                                                                         DBG=MAC      324     A
DBG=MAC   #NUSE    ELSE                                                             DBG=MAC      325     A
DBG=MAC   RTN      BSS    0                                                         DBG=MAC      326     A
DBG=MAC   #NUSE    ENDIF                                                            DBG=MAC      327     A
DBG=MAC  *                                                                         DBG=MAC      328     A
DBG=MAC   SNAP     ENDM                                                             DBG=MAC      329     A
DBG=MAC   SNG=     SPACE  4,8                                                       DBG=MAC      330     A
DBG=MAC  **        SNG= - GENERATE INDIRECT/ADDRESS FIELDS.                         DBG=MAC      331     A
DBG=MAC  *                                                                         DBG=MAC      332     A
DBG=MAC  *                                                                         DBG=MAC      333     A
DBG=MAC  *         GENERATES--                                                     DBG=MAC      334     A
DBG=MAC  *                                                                         DBG=MAC      335     A
DBG=MAC  *         VFD    18/0                                                     DBG=MAC      336     A
DBG=MAC  *         VFD    21/NR OF LEVELS OF INDIRECT ADDRESSING (0=DIRECT)        DBG=MAC      337     A
DBG=MAC  *         VFD    21/ADDRESS AT BEGINNING OF INDIRECT CHAIN                DBG=MAC      338     A
DBG=MAC  *                                                                         DBG=MAC      339     A
DBG=MAC  *         SNG=   (*ADDR)                                                  DBG=MAC      340     A
DBG=MAC  *                                                                         DBG=MAC      341     A
DBG=MAC  *         ENTRY  *        =  A LEVEL OF INDIRECT ADDRESSING.              DBG=MAC      342     A
DBG=MAC  *                              NO * MEANS DIRECT ADDRESSING.              DBG=MAC      343     A
DBG=MAC  *                              E.G. *ADDR MEANS THE CONTENTS OF ADDR      DBG=MAC      344     A
DBG=MAC  *                              CONTAIN AN ADDRESS THAT IS TO BE USED BY   DBG=MAC      345     A
DBG=MAC  *                              SNAP UTILITY.                              DBG=MAC      346     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=MAC

```
DBG=MAC  *                ADDR     =  THE BEGINNING OF THE ADDRESS CHAIN.                DBG=MAC     347     A
DBG=MAC                                                                                   DBG=MAC     348     A
DBG=MAC                                                                                   DBG=MAC     349     A
DBG=MAC            PURGMAC SNG=                                                           DBG=MAC     350     A
DBG=MAC                                                                                   DBG=MAC     351     A
DBG=MAC   SNG=     MACRO   ARG                                                            DBG=MAC     352     A
DBG=MAC   .9       SET     -1                                                             DBG=MAC     353     A
DBG=MAC   +        VFD     18/0                                                           DBG=MAC     354     A
DBG=MAC  *                                                                                DBG=MAC     355     A
DBG=MAC   .D       DUP     -1                                                             DBG=MAC     356     A
DBG=MAC   .9       SET     .9+1                                                           DBG=MAC     357     A
DBG=MAC   .1       MICRO   .9+1,1,/ARG/                                                   DBG=MAC     358     A
DBG=MAC            IFC     NE,/".1"/*/,1                                                  DBG=MAC     359     A
DBG=MAC            STOPDUP                                                                DBG=MAC     360     A
DBG=MAC   .D       ENDD                                                                   DBG=MAC     361     A
DBG=MAC  *                                                                                DBG=MAC     362     A
DBG=MAC   .2       MICRO   .9+1,,/ARG/                                                    DBG=MAC     363     A
DBG=MAC  *                                                                                DBG=MAC     364     A
DBG=MAC   #RG      IF      REG,".2"                                                       DBG=MAC     365     A
DBG=MAC            IFEQ    .9,0,1                                                         DBG=MAC     366     A
DBG=MAC   .9       SET     .9+1                                                           DBG=MAC     367     A
DBG=MAC   .3       MICRO   1,1,/".2"/                                                     DBG=MAC     368     A
DBG=MAC   .4       MICRO   2,1,/".2"/                                                     DBG=MAC     369     A
DBG=MAC   .2       MICRO   1,,/=XSV=".3"+".4"/                                            DBG=MAC     370         I
         -CCGA017
DBG=MAC   .2       MICRO   1,,/=XIDP=SV_".3"+".4"/                                        CCGA017       7     A
DBG=MAC  *                                                                                DBG=MAC     371     A
DBG=MAC   #RG      ELSE                                                                   DBG=MAC     372     A
DBG=MAC   .5       MICRO   1,2,/".2"/                                                     DBG=MAC     373     A
DBG=MAC            IFC     EQ,/".5"/=X/,1                                                 DBG=MAC     374     A
DBG=MAC   .2       MICRO   3,,/".2"/                                                      DBG=MAC     375     A
DBG=MAC            IF      -DEF,".2",1                                                    DBG=MAC     376     A
DBG=MAC            SKIP    1                                                              DBG=MAC     377     A
DBG=MAC            IF      -ABS,".2",1                                                    DBG=MAC     378     A
DBG=MAC   .2       MICRO   1,,/=X".2"/                                                    DBG=MAC     379     A
DBG=MAC   #RG      ENDIF                                                                  DBG=MAC     380     A
DBG=MAC  *                                                                                DBG=MAC     381     A
DBG=MAC            VFD     21/.9                                                          DBG=MAC     382     A
DBG=MAC            VFD     21/".2"                                                        DBG=MAC     383     A
DBG=MAC   SNG=     ENDM                                                                   DBG=MAC     384     A
DBG=MAC   USF=     SPACE   4,8                                                  000390    DBG=MAC     385     A
DBG=MAC  **        USF= - GENERATE USER FLAG PARAMETER CELL.                    000400    DBG=MAC     386     A
DBG=MAC  *                                                                      000410    DBG=MAC     387     A
DBG=MAC  *                                                                      000420    DBG=MAC     388     A
DBG=MAC  *         THIS IS A DUMMY TO CREATE A ZERO WORD IF THE USER            000430    DBG=MAC     389     A
DBG=MAC  *         DOES NOT DESIRE OWN CODE.                                    000440    DBG=MAC     390     A
DBG=MAC  *                                                                      000450    DBG=MAC     391     A
DBG=MAC  *         USF=    USF                                                  000460    DBG=MAC     392     A
DBG=MAC  *                                                                      000470    DBG=MAC     393     A
DBG=MAC  *         ENTRY   USF  = ANY LEGAL STRING OF CHARACTERS                000480    DBG=MAC     394     A
DBG=MAC  *                                                                      000490    DBG=MAC     395     A
DBG=MAC  *         EXIT    NONE                                                 000500    DBG=MAC     396     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=MAC

DBG=MAC  *                                                       000510  DBG=MAC     397    A
DBG=MAC  *         USES   NONE                                   000520  DBG=MAC     398    A
DBG=MAC  *                                                       000530  DBG=MAC     399    A
DBG=MAC  *         CALLS  NONE                                   000540  DBG=MAC     400    A
DBG=MAC                                                          000550  DBG=MAC     401    A
DBG=MAC   .1       IF     -MAC,USF=                              000560  DBG=MAC     402    A
DBG=MAC   USF=     MACRO  USF                                    000570  DBG=MAC     403    A
DBG=MAC            CON    0                                      000580  DBG=MAC     404    A
DBG=MAC   USF=     ENDM                                          000590  DBG=MAC     405    A
DBG=MAC   .1       ENDIF                                         000600  DBG=MAC     406    A
DBG=MAC   DBG=MAC  SPACE  4,8                                            DBG=MAC     407    A
DBG=MAC            BASE   *                                              DBG=MAC     408    A
DBG=MAC   QUAL$    IF     -DEF,QUAL$                                     DBG=MAC     409    A
DBG=MAC            QUAL   *                                              DBG=MAC     410    A
DBG=MAC   QUAL$    ENDIF                                                 DBG=MAC     411    A
DBG=MAC            ENDX                                                  DBG=MAC     412    A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - DBG=MAC

                IDENTIFIER                TOTAL    ACTIVE

                DBG=MAC                     412       410
                CCGA017                       2         2

                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

DBG=IDP  *COMDECK DBG=IDP                                                DBG=IDP       1    A
DBG=IDP   DBG=IDP  CTEXT  DBG=IDP - INTERACTIVE AND BATCH DEBUG PACKAGES.  DBG=IDP       2    A
DBG=IDP   DBG=IDP  SPACE  4,8                                            DBG=IDP       3    A
DBG=IDP            IF     -DEF,QUAL$,1                                   DBG=IDP       4    A
DBG=IDP            QUAL   DBG=IDP                                        DBG=IDP       5    A
DBG=IDP            BASE   D                                              DBG=IDP       6    A
DBG=IDP                                                          000190  DBG=IDP       7    A
DBG=IDP   IDP      MICRO  1,,/"QUAL"/                            000200  DBG=IDP       8    A
DBG=IDP   DBG=IDP  SPACE  4,8                                            DBG=IDP       9    A
DBG=IDP  **        DBG=IDP - INTERACTIVE AND BATCH DEBUG PACKAGES.       DBG=IDP      10    A
DBG=IDP  *                                                               DBG=IDP      11    A
DBG=IDP  *                                                               DBG=IDP      12    A
DBG=IDP  *                CONTAINS THE INTERACTIVE DEBUG PACKAGE (IDP=), WHICH  DBG=IDP      13    A
DBG=IDP  *         PLACES A HOST PROGRAM IN INTERACTIVE MODE; AND THE 2 BATCH  DBG=IDP      14    A
DBG=IDP  *         DEBUGGING ROUTINES (REG=,SNP=), WHICH PROVIDE REGISTER AND  DBG=IDP      15    A
DBG=IDP  *         CORE SNAPSHOT DUMPS.                                  DBG=IDP      16    A
DBG=IDP  *                                                               DBG=IDP      17    A
DBG=IDP  *            IN THE SECTIONS THAT FOLLOW, *ASSEMBLY-TIME DEPENDENCIES*  DBG=IDP      18    A
DBG=IDP  *         ARE DEFINED TO BE THE INTERFACES THAT THE IDP INSTALLER MUST  DBG=IDP      19    A
DBG=IDP  *         PROVIDE; *ASSEMBLY-TIME OPTIONS* ARE THOSE INTERFACES WHICH  DBG=IDP      20    A
DBG=IDP  *         ARE OPTIONAL, I.E. INTERFACES THAT HAVE DEFAULT CONDITIONS.  DBG=IDP      21    A
DBG=IDP  *            IN THE CASE WHERE THE USER PROVIDES A SUBROUTINE THAT WILL  DBG=IDP      22    A
DBG=IDP  *         BE CALLED BY IDP, THE *ENTRY* AND *EXIT* CONDITIONS ARE SHOWN.  DBG=IDP      23    A
DBG=IDP  *         IN ADDITION, THE REGISTERS THAT CANNOT BE ALTERED BY THE USER  DBG=IDP      24    A
DBG=IDP  *         SUBROUTINE ARE SPECIFIED VIA *LOCKED*.  (B1) ARE ASSUMED TO BE  DBG=IDP      25    A
DBG=IDP  *         EQUAL TO 1 (LOCKED), AND WILL THEREFORE NOT BE MENTIUNED  DBG=IDP      26    A

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *         AGAIN.                                                          DBG=IDP      27     A
DBG=IDP  *                                                                         DBG=IDP      28     A
DBG=IDP  ***********************************************************************000153  DBG=IDP      29     A
DBG=IDP  *                                                                 000154  DBG=IDP      30     A
DBG=IDP  *         ASSEMBLY-TIME DEPENDENCIES--                                    DBG=IDP      31     A
DBG=IDP  *                                                                         DBG=IDP      32     A
DBG=IDP  ***********************************************************************000170  DBG=IDP      33     A
DBG=IDP  *                                                                 000180  DBG=IDP      34     A
DBG=IDP  *         COMDECKS NEEDED BY *DBG=IDP*.  ALTHOUGH THESE COMDECKS  000190  DBG=IDP      35     A
DBG=IDP  *         ARE NOT ALL STRICTLY ASSEMBLY-TIME DEPENDENCIES (I.E. SOME CAN000200  DBG=IDP      36     A
DBG=IDP  *         BE SATISFIED AT LOAD-TIME), THE POINT IS THAT THEY ALL HAVE  000210  DBG=IDP      37     A
DBG=IDP  *         TO BE AROUND FOR *DBG=IDP* TO WORK.  SOME ARE THE CLASSIC    000220  DBG=IDP      38     A
DBG=IDP  *         *KRONOS* COMDECKS AND SOME ARE NEW.  ALL CAN BE FOUND ON     000230  DBG=IDP      39     A
DBG=IDP  *         THE *FTN* OLDPL OR ON *CPUTEXT*.  (COMMON COMDECKS ARE AFOOT).000240  DBG=IDP      40     A
DBG=IDP  *                                                                 000250  DBG=IDP      41     A
DBG=IDP  *         WARNING-- IT IS SUGGESTED THAT THE IDP INSTALLER PROVIDE     000260  DBG=IDP      42     A
DBG=IDP  *         SEPARATE COPIES OF CODE COMDECKS FOR *DBG=IDP*, EVEN         000270  DBG=IDP      43     A
DBG=IDP  *         IF THE HOST ALREADY HAS COPIES OF THESE COMDECKS.            000280  DBG=IDP      44     A
DBG=IDP  *         THIS WILL AVOID PROBLEMS THAT CAN OCCUR IN *STEP*            000290  DBG=IDP      45     A
DBG=IDP  *         MODE WHEN IDP TRIES TO STEP COMDECKS THAT IT IS              000300  DBG=IDP      46     A
DBG=IDP  *         USING ITSELF.                                                000310  DBG=IDP      47     A
DBG=IDP  *                                                                 000320  DBG=IDP      48     A
DBG=IDP  *         ACTCOM    COMACIO   COMADEF   COMAFET   COMAREG              000330  DBG=IDP      49     A
DBG=IDP  *                                                                 000340  DBG=IDP      50     A
DBG=IDP  *         COMCCDD   COMCCIO   COMCCOD   COMCDXB   COMCRDC   COMCRDW    000350  DBG=IDP      51     A
DBG=IDP  *         COMCSVR   COMCWTC   COMCWTW                                  000360  DBG=IDP      52     A
DBG=IDP  *                                                                 000370  DBG=IDP      53     A
DBG=IDP  *         COMSRAS                                                      000380  DBG=IDP      54     A
DBG=IDP  ***********************************************************************        DBG=IDP      55     A
DBG=IDP  *                                                                         DBG=IDP      56     A
DBG=IDP  *         ASSEMBLY-TIME OPTIONS--                                         DBG=IDP      57     A
DBG=IDP  *                                                                         DBG=IDP      58     A
DBG=IDP  ***********************************************************************        DBG=IDP      59     A
DBG=IDP  *                                                                 000240  DBG=IDP      60     A
DBG=IDP  *         CP.NFLS - NOMINAL FIELD LENGTH SCM.                          000250  DBG=IDP      61     A
DBG=IDP  *                                                                 000260  DBG=IDP      62     A
DBG=IDP  *         IF THE SYMBOL *CP.NFLS* IS DEFINED (DEF), THEN IT IS THE     000270  DBG=IDP      63     A
DBG=IDP  *         ADDR OF THE SCM LOCATION THAT CONTAINS THE CURRENT NOMINAL   000280  DBG=IDP      64     A
DBG=IDP  *         FIELD LENGTH SCM FOR THE HOST PROGRAM.  THIS OPTION IS       000290  DBG=IDP      65     A
DBG=IDP  *         INTENDED FOR HOST PROGRAMS THAT PERFORM THEIR OWN MEMORY     000300  DBG=IDP      66     A
DBG=IDP  *         MANAGEMENT.  IF *CP.NFLS* IS DEFINED, THEN *IDP* USES ITS    000310  DBG=IDP      67     A
DBG=IDP  *         CONTENTS FOR ADDR LEGALITY CHECKS AND ASSUMES THAT THE HOST  000320  DBG=IDP      68     A
DBG=IDP  *         IS KEEPING IT UPDATED.                                       000330  DBG=IDP      69     A
DBG=IDP  *                                                                 000340  DBG=IDP      70     A
DBG=IDP  *               0 .LE. LEGAL ADDR .LT. (CP.NFLS)                      000350  DBG=IDP      71     A
DBG=IDP  *                                                                 000360  DBG=IDP      72     A
DBG=IDP  *         FORMAT --                                                    000370  DBG=IDP      73     A
DBG=IDP  *               VFD   42/0,18/FL                                       000380  DBG=IDP      74     A
DBG=IDP  *                                                                 000390  DBG=IDP      75     A
DBG=IDP  *         IF THE SYMBOL *CP.NFLS* IS NOT DEFINED (-DEF), THEN *IDP*    000400  DBG=IDP      76     A
DBG=IDP  *         WILL CHECK SCM ADDRESSES USING THE FOLLOWING SCHEME--        000410  DBG=IDP      77     A
DBG=IDP  *                                                                 000420  DBG=IDP      78     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *          1. IF BITS 0 THRU 17 OF (RA.LWP) ARE .MI., THEN *CMM*      000430  DBG=IDP     79    A
DBG=IDP  *             IS ACTIVE.  *IDP* WILL THEREFORE TRY AND FIND THE       000440  DBG=IDP     80    A
DBG=IDP  *             CURRENT FL AT (DABA), I.E. (-(RA.LWP)).  HOWEVER,       000450  DBG=IDP     81    A
DBG=IDP  *             BECAUSE *CMM* IS NOT REAL GOOD AT KEEPING *DABA*        000460  DBG=IDP     82    A
DBG=IDP  *             UPDATED PROPERLY, *IDP* WILL DOUBLE CHECK AN            000470  DBG=IDP     83    A
DBG=IDP  *             ADDR THAT *DABA* INDICATES IS OUT-OF-RANGE BY PERFORMING 000480  DBG=IDP     84    A
DBG=IDP  *             A SYSTEM *MEMORY* (*MEM* REQUEST) TO FIND OUT THE TRUE  000490  DBG=IDP     85    A
DBG=IDP  *             FL.  NOTE-- *IDP* DOES NOT UNCONDITIONALLY CHECK ADDRS  000500  DBG=IDP     86    A
DBG=IDP  *             AGAINST A *MEMORY* FOR EFFICIENCY REASONS.              000510  DBG=IDP     87    A
DBG=IDP  *                                                                     000520  DBG=IDP     88    A
DBG=IDP  *          2. IF BITS 0 THRU 17 OF (RA.LWP) ARE .PL., THEN ADDRESSES  000530  DBG=IDP     89    A
DBG=IDP  *             ARE CHECKED AGAINST A *MEMORY* REQUEST.                 000540  DBG=IDP     90    A
DBG=IDP  *                                                                     000550  DBG=IDP     91    A
DBG=IDP  ***********************************************************************000560  DBG=IDP     92    A
DBG=IDP  *                                                                             DBG=IDP     93    A
DBG=IDP  ***              IF THE SYMBOL *EOS* IS A MICRO NAME (MIC), THEN              DBG=IDP     94    A
DBG=IDP  *         "EOS" IS A MICRO WHICH DEFINES THE INTERACTIVE                      DBG=IDP     95    A
DBG=IDP  *         END-OF-STATEMENT CHARACTER.  THE USE OF THIS CHARACTER ALLOWS       DBG=IDP     96    A
DBG=IDP  *         AN *IDP* USER TO ENTER MORE THAN ONE INTERACTIVE COMMAND ON         DBG=IDP     97    A
DBG=IDP  *         A SINGLE LINE.                                                      DBG=IDP     98    A
DBG=IDP  *                                                                             DBG=IDP     99    A
DBG=IDP  *         DEFAULT IS --                                                       DBG=IDP    100    A
DBG=IDP  *                                                                             DBG=IDP    101    A
DBG=IDP  *         EOS    MICRO  1,,/;/                                                DBG=IDP    102    A
DBG=IDP  *                                                                             DBG=IDP    103    A
DBG=IDP  *         E.G.   SNAP,100;REG,X1;END                                          DBG=IDP    104    A
DBG=IDP  *                                                                             DBG=IDP    105    A
DBG=IDP  *         NOTE   THE END OF STATEMENT LOGIC WILL NOT WORK PROPERLY IF         DBG=IDP    106    A
DBG=IDP  *                "EOS" IS ONE OF THE FOLLOWING 8 CHARACTERS --                DBG=IDP    107    A
DBG=IDP  *                JKLMNOPQ                                                     DBG=IDP    108    A
DBG=IDP  ***********************************************************************      DBG=IDP    109    A
DBG=IDP  *                                                                             DBG=IDP    110    A
DBG=IDP  ***          IF THE SYMBOL *FAA=* IS DEFINED (DEF), THEN *FAA= - FIND         DBG=IDP    111    A
DBG=IDP  *         ABSOLUTE ADDRESS* IS A USER SUBROUTINE THAT WILL PROVIDE            DBG=IDP    112    A
DBG=IDP  *         *DBG=IDP* WITH AN ABSOLUTE ADDRESS ASSOCIATED WITH A *NAME*.        DBG=IDP    113    A
DBG=IDP  *         THIS OPTION IS INTENDED FOR HOST PROGRAMS THAT CONSIST OF           DBG=IDP    114    A
DBG=IDP  *         MULTIPLE *DECK*S OR *IDENT*S, AND WHERE USERS OF *IDP* WOULD        DBG=IDP    115    A
DBG=IDP  *         LIKE TO BE ABLE TO REFERENCE ADDRESSES RELATIVE TO A                DBG=IDP    116    A
DBG=IDP  *         DECK NAME--  E.G. SNAP NAME+20,,10                                  DBG=IDP    117    A
DBG=IDP  *            IF *FAA=* IS DEFINED, THEN *IDP* WILL *RJ =XFAA=* TO TRY TO      DBG=IDP    118    A
DBG=IDP  *         ASSOCIATE AN ABSOLUTE ADDRESS WITH *NAME* BEFORE HE SEARCHS         DBG=IDP    119    A
DBG=IDP  *         THE USER *SET* TABLE.                                               DBG=IDP    120    A
DBG=IDP  *                                                                             DBG=IDP    121    A
DBG=IDP  **        FAA= - FIND ABSOLUTE ADDRESS.                                       DBG=IDP    122    A
DBG=IDP  *                                                                             DBG=IDP    123    A
DBG=IDP  *         ENTRY  (X0) = MX0 7*CHAR     (CHAR=6)                               DBG=IDP    124    A
DBG=IDP  *                (X1) = 42/0LNAME, 18/0                                       DBG=IDP    125    A
DBG=IDP  *                                                                             DBG=IDP    126    A
DBG=IDP  *         EXIT   (X6) = .PL. IF ABS ADDR ASSOCIATED WITH NAME                 DBG=IDP    127    A
DBG=IDP  *                       .MI. IF NO ADDR KNOWN FOR *NAME* (I.E. NO FIND)       DBG=IDP    128    A
DBG=IDP  *                                                                             DBG=IDP    129    A
DBG=IDP  *         LOCKED X - 0,1                                                      DBG=IDP    130    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  ***********************************************************************  DBG=IDP   131   A
DBG=IDP  *                                                                        DBG=IDP   132   A
DBG=IDP  ***          IF THE SYMBOL *FRA=* IS DEFINED (DEF), THEN *FRA= - FIND    DBG=IDP   133   A
DBG=IDP  *         RELATIVE ADDRESS* IS A USER SUBROUTINE THAT WILL PROVIDE       DBG=IDP   134   A
DBG=IDP  *         *DBG=IDP* WITH A DISPLAY CODE (DPC) NAME AND RELATIVE OFFSET   DBG=IDP   135   A
DBG=IDP  *         ASSOCIATED WITH AN ABSOLUTE ADDRESS. THIS ROUTINE IS THE       DBG=IDP   136   A
DBG=IDP  *         LOGICAL COMPLEMENT TO *FAA=*, AND IS USED BY ROUTINES IN       DBG=IDP   137   A
DBG=IDP  *         *DBG=IDP* TO OUTPUT A MORE HUMAN READABLE ADDRESS FORMAT.      DBG=IDP   138   A
DBG=IDP  *                                                                        DBG=IDP   139   A
DBG=IDP  *         E.G. IN RESPONSE TO *IDP* COMMAND-- CODE NAME+10               DBG=IDP   140   A
DBG=IDP  *         1025   010001023         RJ    1023     6 IN NAME              DBG=IDP   141   A
DBG=IDP  *                                                                        DBG=IDP   142   A
DBG=IDP  *            IF *FRA=* IS DEFINED, WHENEVER A ROUTINE IN *DBG=IDP*       DBG=IDP   143   A
DBG=IDP  *         REQUIRES THIS RELATIVE ADDRESS PLUS OFFSET DPC FORMAT, A       DBG=IDP   144   A
DBG=IDP  *         *EQ =XFRA=* IS EXECUTED.                                       DBG=IDP   145   A
DBG=IDP  *                                                                        DBG=IDP   146   A
DBG=IDP  **        FRA= - FIND RELATIVE ADDRESS.                                  DBG=IDP   147   A
DBG=IDP  *                                                                        DBG=IDP   148   A
DBG=IDP  *         ENTRY  (X1) = 60/ABS ADDR                                      DBG=IDP   149   A
DBG=IDP  *                (B7) = RETURN ADDRESS-- AN *RJ =XFRA=* IS NOT USED      DBG=IDP   150   A
DBG=IDP  *                       BECAUSE IF THE HOST PROGRAM HAS REPRIEVE         DBG=IDP   151   A
DBG=IDP  *                       PROCESSING, THEN *FRA=* COULD BE INCORPORATED    DBG=IDP   152   A
DBG=IDP  *                       INTO IT AND EXIST IN AN AREA CHECKSUMMED         DBG=IDP   153   A
DBG=IDP  *                       BY *RPV*                                         DBG=IDP   154   A
DBG=IDP  *                                                                        DBG=IDP   155   A
DBG=IDP  *         EXIT   (X6) = 1ST WORD OF DPC RESULT                           DBG=IDP   156   A
DBG=IDP  *                       E.G. (X6) = NNNNNN.IN.     (.=BLANK(55B))        DBG=IDP   157   A
DBG=IDP  *                (X7) = 2ND WORD OF DPC RESULT  (-C- FORMAT)             DBG=IDP   158   A
DBG=IDP  *                       E.G. (X7) = XXXXXXX000     (0=00B)               DBG=IDP   159   A
DBG=IDP  *                                                                        DBG=IDP   160   A
DBG=IDP  *         LOCKED X - 5                                                   DBG=IDP   161   A
DBG=IDP  *                A - 0,5,6,7                                             DBG=IDP   162   A
DBG=IDP  ***********************************************************************  DBG=IDP   163   A
DBG=IDP  *                                                                        DBG=IDP   164   A
DBG=IDP  ***              IF THE SYMBOL *PROMPT* IS A MICRO NAME (MIC), THEN      DBG=IDP   165   A
DBG=IDP  *         "PROMPT" IS A MICRO WHICH DEFINES THE INTERACTIVE PROMPT.      DBG=IDP   166   A
DBG=IDP  *         THIS PROMPT WILL BE ISSUED WHENEVER *IDP* REQUIRES THAT THE    DBG=IDP   167   A
DBG=IDP  *         USER ENTER A COMMAND.                                          DBG=IDP   168   A
DBG=IDP  *                                                                        DBG=IDP   169   A
DBG=IDP  *         DEFAULT IS --                                                  DBG=IDP   170   A
DBG=IDP  *                                                                        DBG=IDP   171   A
DBG=IDP  *         PROMPT MICRO  1,,/>>/                                          DBG=IDP   172   A
DBG=IDP  *                                                                        DBG=IDP   173   A
DBG=IDP  *         NOTE - *IDP* PREFIXES THE "PROMPT" WITH A BLANK (55B) PRINT    DBG=IDP   174   A
DBG=IDP  *                CONTROL CHARACTER.                                      DBG=IDP   175   A
DBG=IDP  ***********************************************************************  DBG=IDP   176   A
DBG=IDP  *                                                                        DBG=IDP   177   A
DBG=IDP  ***          IF THE SYMBOL *UKT=* IS DEFINED (DEF), THEN *UKT=* IS THE   DBG=IDP   178   A
DBG=IDP  *         FWA OF THE USER KEYWORD TABLE. THIS OPTION ALLOWS THE USER TO  DBG=IDP   179   A
DBG=IDP  *         PERFORM HER OR HIS OWN KEYWORD PROCESSING IN INTERACTIVE MODE. DBG=IDP   180   A
DBG=IDP  *         IF *UKT=* IS DEFINED, THEN AFTER *IDP* HAS SEARCHED ITS OWN    DBG=IDP   181   A
DBG=IDP  *         KEYWORD TABLE, IT WILL SEARCH THE USER KEYWORD TABLE AT        DBG=IDP   182   A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *         *UKT=* ET SEQ, AND BRANCH TO THE PROCESSOR ADDRESS IF A FIND          DBG=IDP    183    A
DBG=IDP  *         IS MADE. *UKT=* FORMAT--                                              DBG=IDP    184    A
DBG=IDP  *                                                                               DBG=IDP    185    A
DBG=IDP  * UKT=    BSS    0                                                              DBG=IDP    186    A
DBG=IDP  *         VFD    42/0LKEYWORD, 18/PROCESSOR ADDR                                DBG=IDP    187    A
DBG=IDP  *          .                                                                    DBG=IDP    188    A
DBG=IDP  *          .                                                                    DBG=IDP    189    A
DBG=IDP  *         DATA   0           END OF TABLE MARK                                  DBG=IDP    190    A
DBG=IDP  *                                                                               DBG=IDP    191    A
DBG=IDP  *            AFTER KEYWORD PROCESSING, THE USER SHOULD BRANCH TO EITHER         DBG=IDP    192    A
DBG=IDP  *         *IDP=MN* TO REENTER THE MAIN LOOP, OR TO *IDP=ER* TO ISSUE            DBG=IDP    193    A
DBG=IDP  *         MESSAGE *ERROR--* BEFORE REENTERING MAIN LOOP.                        DBG=IDP    194    A
DBG=IDP  ***********************************************************************         DBG=IDP    195    A
DBG=IDP  *                                                                               DBG=IDP    196    A
DBG=IDP  ***          IF THE SYMBOL *UIO=* IS DEFINED (DEF), THEN *UIO= - USER           DBG=IDP    197    A
DBG=IDP  *         IDP OWNCODE* IS A USER SUBROUTINE THAT IS CALLED AFTER ENTRY          DBG=IDP    198    A
DBG=IDP  *         TO *IDP=*. IT ALLOWS THE USER TO PERFORM HER OWN TASKS BEFORE         DBG=IDP    199    A
DBG=IDP  *         ENTERING INTERACTIVE MODE (E.G. CHECKING A MASTER SNAP FLAG).         DBG=IDP    200    A
DBG=IDP  *                                                                               DBG=IDP    201    A
DBG=IDP  **        UIO= - USER IDP OWNCODE.                                              DBG=IDP    202    A
DBG=IDP  *                                                                               DBG=IDP    203    A
DBG=IDP  *         ENTRY  NONE                                                           DBG=IDP    204    A
DBG=IDP  *                                                                               DBG=IDP    205    A
DBG=IDP  *         EXIT   (X1) = .MI. IF TO ENTER INTERACTIVE MODE                       DBG=IDP    206    A
DBG=IDP  *                     = .PL. IF NOT ENTERING INTERACTIVE MODE THIS TIME         DBG=IDP    207    A
DBG=IDP  *                            I.E. ALL REGISTERS ARE RESTORED AND                DBG=IDP    208    A
DBG=IDP  *                                 CONTROL RETURNS TO CALLER                     DBG=IDP    209    A
DBG=IDP  *                                                                               DBG=IDP    210    A
DBG=IDP  *         LOCKED X - 5                                                          DBG=IDP    211    A
DBG=IDP  ***********************************************************************         DBG=IDP    212    A
DBG=IDP  *                                                                               DBG=IDP    213    A
DBG=IDP  ***       .OS - DEFINE OPERATING SYSTEM.                                        DBG=IDP    214    A
DBG=IDP  *                                                                               DBG=IDP    215    A
DBG=IDP  *         .OS = 1   KRONOS 2 OR NOS 1/TS                                        DBG=IDP    216    A
DBG=IDP  *         .OS = 2   SCOPE 2  (CYBER 76)                                         DBG=IDP    217    A
DBG=IDP  *         .OS = 3   SCOPE 3 OR NOS 1/BE  (DEFAULT)                              DBG=IDP    218    A
DBG=IDP  ***********************************************************************         DBG=IDP    219    A
DBG=IDP  *                                                                               DBG=IDP    220    A
DBG=IDP  ***          IF THE SYMBOL *URO=* IS DEFINED (DEF), THEN *URO= - USER           DBG=IDP    221    A
DBG=IDP  *         REG= OWNCODE* IS A USER SUBROUTINE THAT IS CALLED AFTER ENTRY         DBG=IDP    222    A
DBG=IDP  *         TO THE REGISTER SNAPSHOT ROUTINE *REG=* (CALLED VIA *REG*             DBG=IDP    223    A
DBG=IDP  *         MACRO IN *DBG=MAC*). IT ALLOWS THE USER TO PERFORM HIS OWN            DBG=IDP    224    A
DBG=IDP  *         TASKS BEFORE THE REGISTER SNAPSHOT IS TAKEN (E.G. CHECKING A          DBG=IDP    225    A
DBG=IDP  *         MASTER SNAP FLAG).                                                    DBG=IDP    226    A
DBG=IDP  *                                                                               DBG=IDP    227    A
DBG=IDP  **        URO= - USER REG= OWNCODE.                                             DBG=IDP    228    A
DBG=IDP  *                                                                               DBG=IDP    229    A
DBG=IDP  *         ENTRY  (X5)+SN=URF = ADDRESS OF USER FLAGS                  000970    DBG=IDP    230    A
DBG=IDP  *                                                                               DBG=IDP    231    A
DBG=IDP  *         EXIT   (X1) = .MI. IF TO PERFORM REGISTER SNAPSHOT                    DBG=IDP    232    A
DBG=IDP  *                     = .PL. IF NO REGISTER SNAPSHOT THIS TIME                  DBG=IDP    233    A
DBG=IDP  ***********************************************************************         DBG=IDP    234    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *                                                                  DBG=IDP    235    A
DBG=IDP  ***          IF THE SYMBOL *USO=* IS DEFINED (DEF), THEN *USO= - USER  DBG=IDP    236    A
DBG=IDP  *         SNP= OWNCODE* IS A USER SUBROUTINE THAT IS CALLED AFTER ENTRY  DBG=IDP    237    A
DBG=IDP  *         TO THE CORE SNAPSHOT ROUTINE *SNP=* (CALLED VIA *SNAP* MACRO  DBG=IDP    238    A
DBG=IDP  *         IN *DBG=MAC*). IT ALLOWS THE USER TO PERFORM HIS OWN TASKS  DBG=IDP    239    A
DBG=IDP  *         BEFORE THE SNAPSHOT IS TAKEN (E.G. CHECKING A MASTER SNAP  DBG=IDP    240    A
DBG=IDP  *         FLAG).                                                   DBG=IDP    241    A
DBG=IDP  *                                                                  DBG=IDP    242    A
DBG=IDP  **        USO= - USER SNP= OWNCODE.                                DBG=IDP    243    A
DBG=IDP  *                                                                  DBG=IDP    244    A
DBG=IDP  *         ENTRY  (X5)+SN=USF = ADDRESS OF USER FLAGS          000990  DBG=IDP    245    A
DBG=IDP  *                                                                  DBG=IDP    246    A
DBG=IDP  *         EXIT   (X1) = .MI. IF SNAP TO BE TAKEN                   DBG=IDP    247    A
DBG=IDP  *                     = .PL. IF NO SNAP THIS TIME                  DBG=IDP    248    A
DBG=IDP  ************************************************************************  DBG=IDP    249    A
DBG=IDP  *                                                                  DBG=IDP    250    A
DBG=IDP  ***             AS A SPACE OPTIMIZATION, IF THE FOLLOWING GENERAL  DBG=IDP    251    A
DBG=IDP  *         UTILITY SUBROUTINES ARE DEFINED (DEF), THEN THEY WILL NOT BE  DBG=IDP    252    A
DBG=IDP  *         ASSEMBLED WITHIN *DBG=IDP*--                             DBG=IDP    253    A
DBG=IDP  *                                                                  DBG=IDP    254    A
DBG=IDP  *         WOD - CONVERT A FULL BINARY WORD TO OCTAL DPC.           DBG=IDP    255    A
DBG=IDP  *                                                                  DBG=IDP    256    A
DBG=IDP  *               THIS OPTIOM APPLIES MAINLY TO HOST PROGRAMS THAT HAVE  DBG=IDP    257    A
DBG=IDP  *         *DBG=IDP* ASSEMBLED ONLY IN A TEST OR DEBUGGING CONFIGURATION,  DBG=IDP    258    A
DBG=IDP  *         BUT THAT REQUIRE THE ABOVE UTILITY SUBROUTINES UNDER ALL  DBG=IDP    259    A
DBG=IDP  *         CONFIGURATIONS, THEREBY NECESSITATING A SEPARATE COPY OF THEM.  DBG=IDP    260    A
DBG=IDP  ************************************************************************  DBG=IDP    261    A
DBG=IDP   .OS      SPACE  4,8                                               DBG=IDP    262    A
DBG=IDP  **        DEFINE DEFAULT OPERATING SYSTEM.                         DBG=IDP    263    A
DBG=IDP            IF     -DEF,.OS,1                                        DBG=IDP    264    A
DBG=IDP   .OS      =      3           SCOPE 3                               DBG=IDP    265    A
DBG=IDP   CHARMX   SPACE  4,8                                               DBG=IDP    266    A
DBG=IDP  **        CHARMX - GENERATE CHARACTER SHIFT MASK WORD.             DBG=IDP    267    A
DBG=IDP  *                                                                  DBG=IDP    268    A
DBG=IDP  *                                                                  DBG=IDP    269    A
DBG=IDP  * LAB     CHARMX CHAR,BIAS                                         DBG=IDP    270    A
DBG=IDP  *                                                                  DBG=IDP    271    A
DBG=IDP  * MX=NR09 CHARMX (0,1,2,3,4,5,6,7,8,9)                             DBG=IDP    272    A
DBG=IDP  * MX=TOKN CHARMX ("EOS",+,-,*,(,)),+7777B-1RM                      DBG=IDP    273    A
DBG=IDP  *                                                                  DBG=IDP    274    A
DBG=IDP  *         ENTRY  LAB  = COMPASS ADDRESS FIELD OF CHARACTER SHIFT MASK  DBG=IDP    275    A
DBG=IDP  *                        WORD                                      DBG=IDP    276    A
DBG=IDP  *                CHAR = A LIST OF CHARACTERS TO GENERATE SHIFT MASK FOR  DBG=IDP    277    A
DBG=IDP  *                BIAS = CHAR BIAS (MODULO 60D)                     DBG=IDP    278    A
DBG=IDP  *                                                                  DBG=IDP    279    A
DBG=IDP  *                        IF (LAB) ARE LEFT SHIFTED *CHAR_BIAS*, THEN  DBG=IDP    280    A
DBG=IDP  *                        SIGN BIT WILL BE ON FOR ANY CHARACTER THAT  DBG=IDP    281    A
DBG=IDP  *                        APPEARED IN *CHAR* LIST, AND OFF FOR ALL  DBG=IDP    282    A
DBG=IDP  *                        OTHERS.                                   DBG=IDP    283    A
DBG=IDP  *                                                                  DBG=IDP    284    A
DBG=IDP  *         EXIT   NONE                                              DBG=IDP    285    A
DBG=IDP  *                                                                  DBG=IDP    286    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *         USES   NONE                                                    DBG=IDP     287     A
DBG=IDP  *                                                                        DBG=IDP     288     A
DBG=IDP  *         CALLS  NONE                                                    DBG=IDP     289     A
DBG=IDP                                                                           DBG=IDP     290     A
DBG=IDP            NOREF  .2,.3,.4                                                DBG=IDP     291     A
DBG=IDP                                                                           DBG=IDP     292     A
DBG=IDP            PURGMAC CHARMX                                                 DBG=IDP     293     A
DBG=IDP                                                                           DBG=IDP     294     A
DBG=IDP            MACRO  CHARMX,LAB,CHAR,BIAS                                    DBG=IDP     295     A
DBG=IDP  *                                                                        DBG=IDP     296     A
DBG=IDP   .1       MICRO  1,60,/000000000000000000000000000000000000000000000000  DBG=IDP     297     A
DBG=IDP  ,00000000000/                                                            DBG=IDP     298     A
DBG=IDP  *                                                                        DBG=IDP     299     A
DBG=IDP   CHAR     IRP    CHAR                                                    DBG=IDP     300     A
DBG=IDP   .2       MICRO  1,,/CHAR/                                               DBG=IDP     301     A
DBG=IDP   .3       MICCNT .2                                                      DBG=IDP     302     A
DBG=IDP  *                                                                        DBG=IDP     303     A
DBG=IDP            IFEQ   .3,1,2                                                  DBG=IDP     304     A
DBG=IDP   .4       SET    1R".2"                                                  DBG=IDP     305     A
DBG=IDP            SKIP   1                                                       DBG=IDP     306     A
DBG=IDP   .4       SET    ".2"                                                    DBG=IDP     307     A
DBG=IDP  *                                                                        DBG=IDP     308     A
DBG=IDP            IFC    NE,/BIAS//,1                                            DBG=IDP     309     A
DBG=IDP   .4       SET    .4_BIAS                                                 DBG=IDP     310     A
DBG=IDP  *                                                                        DBG=IDP     311     A
DBG=IDP   .4       OCTMIC .4,2                                                    DBG=IDP     312     A
DBG=IDP   .2       SET    ".4"B                                                   DBG=IDP     313     A
DBG=IDP  *                                                                        DBG=IDP     314     A
DBG=IDP            IFNE   .2,0,2                                                  DBG=IDP     315     A
DBG=IDP   .3       MICRO  1,.2,/".1"/                                             DBG=IDP     316     A
DBG=IDP            SKIP   1                                                       DBG=IDP     317     A
DBG=IDP   .3       MICRO  1,,//                                                   DBG=IDP     318     A
DBG=IDP  *                                                                        DBG=IDP     319     A
DBG=IDP            IFNE   .2,59,2                                                 DBG=IDP     320     A
DBG=IDP   .4       MICRO  .2+2,,/".1"/                                            DBG=IDP     321     A
DBG=IDP            SKIP   1                                                       DBG=IDP     322     A
DBG=IDP   .4       MICRO  1,,//                                                   DBG=IDP     323     A
DBG=IDP  *                                                                        DBG=IDP     324     A
DBG=IDP   .1       MICRO  1,60,/".3"1".4"/                                        DBG=IDP     325     A
DBG=IDP   CHAR     IRP                                                            DBG=IDP     326     A
DBG=IDP  *                                                                        DBG=IDP     327     A
DBG=IDP   .2       SET    0                                                       DBG=IDP     328     A
DBG=IDP   LAB      BSS    0                                                       DBG=IDP     329     A
DBG=IDP  *                                                                        DBG=IDP     330     A
DBG=IDP   BIT      DUP    60                                                      DBG=IDP     331     A
DBG=IDP   .2       SET    .2+1                                                    DBG=IDP     332     A
DBG=IDP   .3       MICRO  .2,1,/".1"/                                             DBG=IDP     333     A
DBG=IDP  *                                                                        DBG=IDP     334     A
DBG=IDP            POS    60-.2+1                                                 DBG=IDP     335     A
DBG=IDP            VFD    1/".3"                                                  DBG=IDP     336     A
DBG=IDP   BIT      ENDD                                                           DBG=IDP     337     A
DBG=IDP  *                                                                        DBG=IDP     338     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP   CHARMX   ENDM                                                    DBG=IDP     339    A
DBG=IDP   LXQ      SPACE  4,8                                              DBG=IDP     340    A
DBG=IDP  **        LXQ - REDEFINE THE LEFT SHIFT INSTRUCTION.              DBG=IDP     341    A
DBG=IDP  *                                                                 DBG=IDP     342    A
DBG=IDP  *         THIS OPDEF REDEFINES THE LEFT SHIFT INSTRUCTION TO SUPPRESS  DBG=IDP  343    A
DBG=IDP  *         CODE GENERATION WHEN THE SHIFT COUNT IS 0, +60D OR -60D.  DBG=IDP     344    A
DBG=IDP  *         THE INSTRUCTION IS OTHERWISE UNCHANGED.                 DBG=IDP     345    A
DBG=IDP  *                                                                 DBG=IDP     346    A
DBG=IDP  *         LXI       JK                                            DBG=IDP     347    A
DBG=IDP  *                                                                 DBG=IDP     348    A
DBG=IDP  *         ENTRY  *XI* = X-REG TO BE SHIFTED                       DBG=IDP     349    A
DBG=IDP  *                *JK* = SHIFT COUNT EXPRESSION                    DBG=IDP     350    A
DBG=IDP  *                                                                 DBG=IDP     351    A
DBG=IDP  *         USES   XI                                               DBG=IDP     352    A
DBG=IDP                                                                    DBG=IDP     353    A
DBG=IDP                                                                    DBG=IDP     354    A
DBG=IDP            PURGDEF ^XQ                                             DBG=IDP     355    A
DBG=IDP            PURGDEF   LXQ                                           DBG=IDP     356    A
DBG=IDP   ^XQ      CPOP   0,200B,100B                                      DBG=IDP     357    A
DBG=IDP                                                                    DBG=IDP     358    A
DBG=IDP   LXQ      OPDEF     I,JK                                          DBG=IDP     359    A
DBG=IDP    IFNE JK,0,2                                                     DBG=IDP     360    A
DBG=IDP    IFNE JK_&60D,0,1                                                DBG=IDP     361    A
DBG=IDP    ^X.I JK                                                         DBG=IDP     362    A
DBG=IDP    ENDM                                                            DBG=IDP     363    A
DBG=IDP   PRIDP    SPACE  4,8                                              DBG=IDP     364    A
DBG=IDP  **        PRIDP - PRINT CODED LINE ON OUTPUT FILE.                DBG=IDP     365    A
DBG=IDP  *                                                                 DBG=IDP     366    A
DBG=IDP  *                                                                 DBG=IDP     367    A
DBG=IDP  *         PRIDP  FWA,LEN,NRB                                      DBG=IDP     368    A
DBG=IDP  *                                                                 DBG=IDP     369    A
DBG=IDP  *         ENTRY  FWA = FWA OF LINE (-C- FORMAT)                   DBG=IDP     370    A
DBG=IDP  *                LEN = LENGTH OF LINE (IN WORDS)                  DBG=IDP     371    A
DBG=IDP  *                    = .ZR. IF LENGTH TO BE COMPUTED              DBG=IDP     372    A
DBG=IDP  *                NRB = NR OF BLANK LINES TO OUTPUT PRECEDING THE LINE  DBG=IDP  373    A
DBG=IDP  *                                                                 DBG=IDP     374    A
DBG=IDP  *         EXIT   NONE                                             DBG=IDP     375    A
DBG=IDP  *                                                                 DBG=IDP     376    A
DBG=IDP  *         USES   ALL BUT B4,A0,X0,A5,X5 (INCLUDES ALL CALLS)      DBG=IDP     377    A
DBG=IDP  *                                                                 DBG=IDP     378    A
DBG=IDP  *         CALLS  ROL                                              DBG=IDP     379    A
DBG=IDP                                                                    DBG=IDP     380    A
DBG=IDP                                                                    DBG=IDP     381    A
DBG=IDP            PURGMAC PRIDP                                           DBG=IDP     382    A
DBG=IDP                                                                    DBG=IDP     383    A
DBG=IDP   PRIDP    MACRO  FWA,LEN,NRB                                      DBG=IDP     384    A
DBG=IDP            =X6    FWA                                              DBG=IDP     385    A
DBG=IDP            =X7    LEN                                              DBG=IDP     386    A
DBG=IDP            =X4    NRB                                              DBG=IDP     387    A
DBG=IDP            RJ     =XROL                                            DBG=IDP     388    A
DBG=IDP   PRIDP    ENDM                                                    DBG=IDP     389    A
DBG=IDP   PRBDO    SPACE  4,8                                              DBG=IDP     390    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  **        PRBDO - PRINT CODED LINE ON BATCH DEBUG OUTPUT FILE.
DBG=IDP  *
DBG=IDP  *
DBG=IDP  *                 IF THE USER DID NOT DEFINE A BATCH DEBUG OUTPUT FILE,
DBG=IDP  *         (F.BDO -DEF), THEN *IDP* WILL INVENT ONE AND ACCESS IT VIA
DBG=IDP  *         *PRBDO* MACRO. IF *F.BDO* IS DEFINED, THEN THE USER SUPPLIES
DBG=IDP  *         HER OWN *PRBDO*.
DBG=IDP  *
DBG=IDP  *         PRBDO  FWA,LEN
DBG=IDP  *
DBG=IDP  *         ENTRY  FWA = FWA OF LINE (-C- FORMAT)
DBG=IDP  *                LEN = LENGTH OF LINE (IN WORDS)
DBG=IDP  *                    = .ZR. IF LENGTH TO BE COMPUTED
DBG=IDP  *
DBG=IDP  *         EXIT   NONE
DBG=IDP  *
DBG=IDP  *         USES   ALL BUT A0,X0,A5,X5   (INCLUDES ALL CALLS)
DBG=IDP  *
DBG=IDP  *         CALLS  WRITEC
DBG=IDP
DBG=IDP
DBG=IDP   #BDO     IF     -DEF,F.BDO
DBG=IDP   #OS      IFNE   .OS,2       IF NOT SCOPE 2
DBG=IDP
DBG=IDP            PURGMAC PRBDO
DBG=IDP
DBG=IDP   PRBDO    MACRO  FWA,LEN
DBG=IDP            WRITEC =XF.BDO,(FWA),(LEN)
DBG=IDP   PRBDO    ENDM
DBG=IDP   #OS      ENDIF
DBG=IDP   #BDO     ENDIF
DBG=IDP   MICROS   SPACE  4,8
DBG=IDP  **        DEFINE DEFAULT MICROS.
DBG=IDP
DBG=IDP            IFNE   .OS,2,2
DBG=IDP            IF     -MIC,EOS,1
DBG=IDP   EOS      MICRO  1,,/;/
DBG=IDP
DBG=IDP            IFNE   .OS,2,2
DBG=IDP            IF     -MIC,PROMPT,1
DBG=IDP   PROMPT   MICRO  1,,/>>/
DBG=IDP   DEFS     SPACE  4,8
DBG=IDP  **        SYMBOL DEFINITIONS.
DBG=IDP
DBG=IDP
DBG=IDP   CHAR     =      6           LENGTH OF CYBER CHARACTER
DBG=IDP
DBG=IDP   L.BSL    =      4           NR OF CM WORDS DUMPED ON LINE (BATCH)
DBG=IDP   L.ISL    =      2           NR OF CM WORDS DUMPED ON LINE (INTERACTIVE)
DBG=IDP
DBG=IDP   L.PRB    =      0           NR OF PRECEDING BLANKS TO OUTPUT
DBG=IDP  *                              FOR SPACING PURPOSES
```

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP   DEFS     SPACE  4,8                                                    DBG=IDP     443    A
DBG=IDP  **        SNAP PARAMETER LIST SYMBOL DEFINITIONS.                       DBG=IDP     444    A
DBG=IDP  *           THESE SYMBOLS DEFINE THE ORDINALS INTO THE SNAP PARAMETER   DBG=IDP     445    A
DBG=IDP  *           LIST. (DESIRED WORD IS AT-- FWA OF PARAMETER LIST + SN=XXX) DBG=IDP     446    A
DBG=IDP                                                                          DBG=IDP     447    A
DBG=IDP                                                                          DBG=IDP     448    A
DBG=IDP   SN=FRK   =      0          1ST WORD OF FREQUENCY PARAMETER LIST        DBG=IDP     449    A
DBG=IDP   SN=LL    =      0          LOWER LIMIT                                 DBG=IDP     450    A
DBG=IDP   SN=UL    =      1          UPPER LIMIT                                 DBG=IDP     451    A
DBG=IDP   SN=INC   =      2          INCREMENT                                   DBG=IDP     452    A
DBG=IDP   SN=HDR   =      3          HEADER                                      DBG=IDP     453    A
DBG=IDP   SN=CNT   =      4          SNAP COUNT                                  DBG=IDP     454    A
DBG=IDP   SN=FWA   =      5          FWA                                         DBG=IDP     455    A
DBG=IDP   SN=LWA   =      6          LWA                                         DBG=IDP     456    A
DBG=IDP   SN=LEN   =      7          LENGTH                                      DBG=IDP     457    A
DBG=IDP   SN=URF   =      5          USER *REG* MACRO FLAGS                 001010 DBG=IDP     458    A
DBG=IDP   SN=RRL   =      6          1ST WORD OF *RGR=* LIST FOR *REG* MACRO 001020 DBG=IDP     459    A
DBG=IDP   SN=USF   =      8          USER *SNAP* MACRO FLAGS                001030 DBG=IDP     460    A
DBG=IDP   SN=SRL   =      9          1ST WORD OF *RGR=* LIST FOR *SNAP* MACRO 001040 DBG=IDP     461    A
DBG=IDP   APL      SPACE  4,8                                                    DBG=IDP     462    A
DBG=IDP  *         APL - DUMMY APLIST USED FOR SAVE CELLS AND *IDP* CALLS.       DBG=IDP     463    A
DBG=IDP                                                                          DBG=IDP     464    A
DBG=IDP   APL      BSSZ   SN=LEN+1                                               DBG=IDP     465    A
DBG=IDP                                                                          DBG=IDP     466    A
DBG=IDP            LOC    APL                                                    DBG=IDP     467    A
DBG=IDP                                                                          DBG=IDP     468    A
DBG=IDP   AP=LL    =      *+SN=LL                                               DBG=IDP     469    A
DBG=IDP   AP=UL    =      *+SN=UL                                               DBG=IDP     470    A
DBG=IDP   AP=INC   =      *+SN=INC                                              DBG=IDP     471    A
DBG=IDP   AP=HDR   =      *+SN=HDR                                              DBG=IDP     472    A
DBG=IDP   AP=CNT   =      *+SN=CNT                                              DBG=IDP     473    A
DBG=IDP   AP=FWA   =      *+SN=FWA                                              DBG=IDP     474    A
DBG=IDP   AP=LWA   =      *+SN=LWA                                              DBG=IDP     475    A
DBG=IDP   AP=LEN   =      *+SN=LEN                                              DBG=IDP     476    A
DBG=IDP            LOC    *O                                                    DBG=IDP     477    A
DBG=IDP   DATA     SPACE  4,8                                                    DBG=IDP     478    A
DBG=IDP   FWAPARM  BSSZ   1          SAVE CELL FOR FWA OF PARAMETER LIST         DBG=IDP     479    A
DBG=IDP                                                                          DBG=IDP     480    A
DBG=IDP                                                                          DBG=IDP     481    A
DBG=IDP   SNAPLNE  BSSZ   15         INTERACTIVE AND BATCH DEBUG OUTPUT LINE     DBG=IDP     482    A
DBG=IDP  *                             IMAGE AREA                                DBG=IDP     483    A
DBG=IDP   IDPFLG   SPACE  4,8                                                    DBG=IDP     484    A
DBG=IDP  **        IDPFLG - INTERACTIVE/BATCH MASTER CONTROL FLAG.               DBG=IDP     485    A
DBG=IDP  *                                                                       DBG=IDP     486    A
DBG=IDP  *                CONTAINS GLOBAL CONTROL INFORMATION THAT IS USED       DBG=IDP     487    A
DBG=IDP  *         THROUGH OUT *DBG=IDP*.  THE FIELDS IN *IDPFLG* ARE DESCRIBED  DBG=IDP     488    A
DBG=IDP  *         BY PAIRS OF SYMBOLS OF THE FORM *IDF.XXXP* AND *IDF.XXXL*,    DBG=IDP     489    A
DBG=IDP  *         WHERE *IDF* IS THE COMMON PREFIX, *XXX* IS THE FIELD          DBG=IDP     490    A
DBG=IDP  *         DESCRIPTOR, *P* DENOTES THE RIGHT-MOST BIT OF THE FIELD, AND  DBG=IDP     491    A
DBG=IDP  *         *L* DENOTES THE LENGTH OF THE FIELD.                          DBG=IDP     492    A
DBG=IDP  *                                                                       DBG=IDP     493    A
DBG=IDP  *                FIELD DESCRIPTIONS (XXX) ARE --                        DBG=IDP     494    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *                                                                  DBG=IDP    495    A
DBG=IDP  *         ADR  = 0  IF ADDRESSES ARE TO BE OUTPUT AS ABSOLUTE      DBG=IDP    496    A
DBG=IDP  *              = 1  IF ADDRESSES ARE TO BE OUTPUT AS DECK RELATIVE DBG=IDP    497    A
DBG=IDP  *                                                                  DBG=IDP    498    A
DBG=IDP  *         BDO  = 1  IF WRITING TO BATCH DEBUG OUTPUT FILE (F.BDO), DBG=IDP    499    A
DBG=IDP  *                     ELSE 0                                       DBG=IDP    500    A
DBG=IDP  *                                                                  DBG=IDP    501    A
DBG=IDP  *         BKO  = 1  IF ECHOING/WRITING INPUT SOURCE LINE IMAGE TO  DBG=IDP    502    A
DBG=IDP  *                     BATCH DEBUG OUTPUT FILE *F.BDO*, ELSE 0      DBG=IDP    503    A
DBG=IDP  *                                                                  DBG=IDP    504    A
DBG=IDP  *         FTO  = 0  IF THIS IS 1ST TIME *IDP* HAS BEEN CALLED, ELSE 1 DBG=IDP    505    A
DBG=IDP  *                                                                  DBG=IDP    506    A
DBG=IDP  *         IDO  = 1  IF WRITING TO INTERACTIVE DEBUG OUTPUT FILE (F.IDO), DBG=IDP    507    A
DBG=IDP  *                     ELSE 0                                       DBG=IDP    508    A
DBG=IDP  *                                                                  DBG=IDP    509    A
DBG=IDP  *         IKO  = 1  IF ECHOING/WRITING INPUT SOURCE LINE IMAGE TO  DBG=IDP    510    A
DBG=IDP  *                     INTERACTIVE DEBUG OUTPUT FILE *F.IDO*, ELSE 0 DBG=IDP    511    A
DBG=IDP  *                                                                  DBG=IDP    512    A
DBG=IDP  *         INP  = 1  IF READING FROM BATCH DEBUG INPUT FILE (F.BDI) DBG=IDP    513    A
DBG=IDP  *                     (I.E. READING FROM AN UNCONNECTED INPUT FILE) DBG=IDP    514    A
DBG=IDP  *              = 0  IF READING FROM INTERACTIVE DEBUG INPUT FILE (F.IDI) DBG=IDP    515    A
DBG=IDP  *                                                                  DBG=IDP    516    A
DBG=IDP  *         SNL  = NR OF CM WORDS TO BE DUMPED (BY *DCM*) ON A SINGLE LINE DBG=IDP    517    A
DBG=IDP  *              = L.BSL  IN BATCH MODE                              DBG=IDP    518    A
DBG=IDP  *              = L.ISL  IN INTERACTIVE MODE                        DBG=IDP    519    A
DBG=IDP  *                                                                  DBG=IDP    520    A
DBG=IDP  *         XEC  = 0  IF THE EXECUTIVE IS *IDP=*                     DBG=IDP    521    A
DBG=IDP  *              = 1  IF THE EXECUTIVE IS *REG=*                     DBG=IDP    522    A
DBG=IDP  *              = 2  IF THE EXECUTIVE IS *SNP=*                     DBG=IDP    523    A
DBG=IDP                                                                     DBG=IDP    524    A
DBG=IDP                                                                     DBG=IDP    525    A
DBG=IDP            DESCRIBE IDF.                                            DBG=IDP    526    A
DBG=IDP   IDO      DEFINE 1                                                 DBG=IDP    527    A
DBG=IDP   IKO      DEFINE 1                                                 DBG=IDP    528    A
DBG=IDP            DEFINE 1                                                 DBG=IDP    529    A
DBG=IDP   BDO      DEFINE 1                                                 DBG=IDP    530    A
DBG=IDP   BKO      DEFINE 1                                                 DBG=IDP    531    A
DBG=IDP            DEFINE 1                                                 DBG=IDP    532    A
DBG=IDP   FTO      DEFINE 1                                                 DBG=IDP    533    A
DBG=IDP   INP      DEFINE 1                                                 DBG=IDP    534    A
DBG=IDP   ADR      DEFINE 1                                                 DBG=IDP    535    A
DBG=IDP   XEC      DEFINE 3                                                 DBG=IDP    536    A
DBG=IDP            DEFINE 30                                                DBG=IDP    537    A
DBG=IDP   SNL      DEFINE 18                                                DBG=IDP    538    A
DBG=IDP                                                                     DBG=IDP    539    A
DBG=IDP   IDPFLG   BSSZ   1                                                 DBG=IDP    540    A
DBG=IDP   PAF.     SPACE  4,8                                       000200  DBG=IDP    541    A
DBG=IDP  **        PAF. - *PATFLG* DESCRIBE/DEFINES.                000210  DBG=IDP    542    A
DBG=IDP                                                             000220  DBG=IDP    543    A
DBG=IDP                                                             000230  DBG=IDP    544    A
DBG=IDP            DESCRIBE PAF.,15                                 000240  DBG=IDP    545    A
DBG=IDP   LWA      DEFINE 1                                         000250  DBG=IDP    546    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            DEFINE 2                                                  000260   DBG=IDP      547     A
DBG=IDP   LEN      DEFINE 1                                                  000270   DBG=IDP      548     A
DBG=IDP            DEFINE 11                                                 000280   DBG=IDP      549     A
DBG=IDP   IDPTB    SPACE  4,8                                                         DBG=IDP      550     A
DBG=IDP  **        IDPTB - COMMAND LINE TOKEN BUFFER DESCRIBE/DEFINES.                DBG=IDP      551     A
DBG=IDP                                                                               DBG=IDP      552     A
DBG=IDP                                                                               DBG=IDP      553     A
DBG=IDP            DESCRIBE TB.                                                       DBG=IDP      554     A
DBG=IDP   LAS      DEFINE 1                                                           DBG=IDP      555     A
DBG=IDP            DEFINE 23                                                          DBG=IDP      556     A
DBG=IDP   PRS      DEFINE 18                                                          DBG=IDP      557     A
DBG=IDP   LEN      DEFINE 18                                                          DBG=IDP      558     A
DBG=IDP   F.BDO    SPACE  4,8                                                         DBG=IDP      559     A
DBG=IDP  **        F.BDO - FET AND BUFFER FOR BATCH DEBUG OUTPUT FILE.                DBG=IDP      560     A
DBG=IDP                                                                               DBG=IDP      561     A
DBG=IDP   #BDO     IF     -DEF,F.BDO                                                  DBG=IDP      562     A
DBG=IDP   #OS      IFNE   .OS,2                                                       DBG=IDP      563     A
DBG=IDP   L.BDO    =      101B        LENGTH OF BATCH OUTPUT BUFFER                   DBG=IDP      564     A
DBG=IDP                                                                               DBG=IDP      565     A
DBG=IDP   F.BDO    BSS    0           ** FWA OF FET **                                DBG=IDP      566     A
DBG=IDP   BDO      FILEC  IDPBDO,L.BDO                                                DBG=IDP      567     A
DBG=IDP   IDPBDO   BSS    L.BDO                                                       DBG=IDP      568     A
DBG=IDP   #OS      ENDIF                                                              DBG=IDP      569     A
DBG=IDP   #BDO     ENDIF                                                              DBG=IDP      570     A
DBG=IDP   CAD      SPACE  4,8                                                 000220   DBG=IDP      571     A
DBG=IDP  **        CAD - CONVERT ADDRESS FROM BINARY TO DPC.                  000230   DBG=IDP      572     A
DBG=IDP  *                                                                    000240   DBG=IDP      573     A
DBG=IDP  *                                                                    000250   DBG=IDP      574     A
DBG=IDP  *         THIS ROUTINE CONVERTS A BINARY ADDRESS (18 BITS) TO DPC    000260   DBG=IDP      575     A
DBG=IDP  *         IN ONE OF 2 WAYS --                                        000270   DBG=IDP      576     A
DBG=IDP  *                                                                    000280   DBG=IDP      577     A
DBG=IDP  *           1. IF *FRA=* (FIND RELATIVE ADDRESS) IS DEFINED .AND.    000290   DBG=IDP      578     A
DBG=IDP  *              *IDPFLG/IDF.ADRP* INDICATES THAT WE ARE TO CONVERT    000300   DBG=IDP      579     A
DBG=IDP  *              THE ADDR TO DECK RELATIVE, THEN *CAD* WILL CALL *FRA=*000310   DBG=IDP      580     A
DBG=IDP  *              TO FIND AND CONVERT THE RELATIVE ADDRESS.             000320   DBG=IDP      581     A
DBG=IDP  *                                                                    000330   DBG=IDP      582     A
DBG=IDP  *           2. IF *FRA=* IS NOT DEFINED .OR. *IDPFLG/IDF.ADRP* INDICATES000340   DBG=IDP      583     A
DBG=IDP  *              THAT WE ARE TO CONVERT THE ADDR TO AN ABSOLUTE ADDR,  000350   DBG=IDP      584     A
DBG=IDP  *              *CAD* WILL CALL *COD* (CONVERT BINARY TO OCTAL DPC)   000360   DBG=IDP      585     A
DBG=IDP  *              TO CONVERT THE ADDR.                                  000370   DBG=IDP      586     A
DBG=IDP  *                                                                    000380   DBG=IDP      587     A
DBG=IDP  *         ENTRY  (X1) = ADDR TO CONVERT, IN BINARY.                  000390   DBG=IDP      588     A
DBG=IDP  *                                                                    000400   DBG=IDP      589     A
DBG=IDP  *         EXIT   (X6) = .NNNNNN...  IF ABSOLUTE CONVERSION           000410   DBG=IDP      590     A
DBG=IDP  *                     = ..+NNNNNN.  IF RELATIVE CONVERSION           000420   DBG=IDP      591     A
DBG=IDP  *                                                                    000430   DBG=IDP      592     A
DBG=IDP  *                       WHERE . = BLANK(55B)                         000440   DBG=IDP      593     A
DBG=IDP  *                             N = OCTAL DIGIT (ADDR IS RIGHT JUSTIFIED 000450   DBG=IDP      594     A
DBG=IDP  *                                 TO BIT 3*CHAR, LEADING 0 SUPPRESSION) 000460   DBG=IDP      595     A
DBG=IDP  *                             + = RELATIVE ADDR INDICATOR            000470   DBG=IDP      596     A
DBG=IDP  *                                                                    000480   DBG=IDP      597     A
DBG=IDP  *         USES   ALL BUT A0,X0,A5,X5,A6   (INCLUDES ALL CALLS)       000490   DBG=IDP      598     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *                                                          000500  DBG=IDP    599   A
DBG=IDP  *         CALLS  COD,FRA=(IF DEF)                          000510  DBG=IDP    600   A
DBG=IDP                                                             000520  DBG=IDP    601   A
DBG=IDP                                                             000530  DBG=IDP    602   A
DBG=IDP   CAD      SUBR               ** ENTRY/EXIT **              000540  DBG=IDP    603   A
DBG=IDP                                                             000550  DBG=IDP    604   A
DBG=IDP  *         CHECK FOR AND CONVERT/FIND A RELATIVE ADDRESS.   000560  DBG=IDP    605   A
DBG=IDP                                                             000570  DBG=IDP    606   A
DBG=IDP   #FRA     IF     DEF,FRA=                                  000580  DBG=IDP    607   A
DBG=IDP            SA2    IDPFLG                                    000590  DBG=IDP    608   A
DBG=IDP            LX2    59-IDF.ADRP                               000600  DBG=IDP    609   A
DBG=IDP            PL     X2,CAD3     IF USER REQUESTED ABS CONVERSION 000610  DBG=IDP    610   A
DBG=IDP                                                             000620  DBG=IDP    611   A
DBG=IDP            SB7    CAD2        (B7) = RETURN ADDR FOR *FRA=*  000630  DBG=IDP    612   A
DBG=IDP            EQ     FRA=        FIND RELATIVE ADDRESS          000640  DBG=IDP    613   A
DBG=IDP                                                             000650  DBG=IDP    614   A
DBG=IDP  *         RETURN FROM *FRA=*.  NEED TO SET UP EXIT CONDITIONS. 000660  DBG=IDP    615   A
DBG=IDP                                                             000670  DBG=IDP    616   A
DBG=IDP   CAD2     SA2    =R.   +.                                  000680  DBG=IDP    617   A
DBG=IDP            MX3    6*CHAR                                    000690  DBG=IDP    618   A
DBG=IDP            BX4    X3*X6       (X4) = NNNNNN0000  (.=BLANK(55B),0=00B) 000700  DBG=IDP    619   A
DBG=IDP            IX6    X4+X2       (X6) = NNNNNN...+  (.=BLANK(55B)) 000710  DBG=IDP    620   A
DBG=IDP            LX6    -3*CHAR     (X6) = ..+NNNNNN.               000712  DBG=IDP    621   A
DBG=IDP            EQ     EXIT.                                       000720  DBG=IDP    622   A
DBG=IDP   #FRA     ENDIF                                              000730  DBG=IDP    623   A
DBG=IDP                                                             000740  DBG=IDP    624   A
DBG=IDP  *         HERE IF CONVERTING TO AN ABSOLUTE ADDRESS.       000750  DBG=IDP    625   A
DBG=IDP                                                             000760  DBG=IDP    626   A
DBG=IDP   CAD3     RJ     COD         CONVERT BINARY TO OCTAL DPC  000770  DBG=IDP    627   A
DBG=IDP            LX6    3*CHAR      (X6) = .NNNNNN...   (.=BLANK(55B)) 000780  DBG=IDP    628   A
DBG=IDP            EQ     EXIT.                                       000790  DBG=IDP    629   A
DBG=IDP   DBG=IDP  TITLE  BATCH AND SHARED ROUTINES.                          DBG=IDP    630   A
DBG=IDP   CHK      SPACE  4,8                                                 DBG=IDP    631   A
DBG=IDP  **        CHK - CHECK CM ADDRESS.                                    DBG=IDP    632   A
DBG=IDP  *                                                                    DBG=IDP    633   A
DBG=IDP  *                                                                    DBG=IDP    634   A
DBG=IDP  *         ENTRY  (B2)   =  ADDRESS TO BE CHECKED                     DBG=IDP    635   A
DBG=IDP  *                                                                    DBG=IDP    636   A
DBG=IDP  *         EXIT   (B2)   =  .MI. IF (B2) WAS BAD                      DBG=IDP    637   A
DBG=IDP  *                       =  UNCHANGED IF OK                           DBG=IDP    638   A
DBG=IDP  *                                                                    DBG=IDP    639   A
DBG=IDP  *         USES   X - 1,2                                             DBG=IDP    640   A
DBG=IDP  *                A - 1                                               DBG=IDP    641   A
DBG=IDP  *                B - NONE    (USES B2 IF .GE. FL)             000580  DBG=IDP    642   A
DBG=IDP  *                                                                    DBG=IDP    643   A
DBG=IDP  *         CALLS  MEMORY                                       000600  DBG=IDP    644   A
DBG=IDP                                                                       DBG=IDP    645   A
DBG=IDP                                                                       DBG=IDP    646   A
DBG=IDP   CHK      SUBR               ** ENTRY/EXIT **                        DBG=IDP    647   A
DBG=IDP                                                             000620  DBG=IDP    648   A
DBG=IDP   #NFLS    IF     DEF,CP.NFLS                               000630  DBG=IDP    649   A
DBG=IDP            SA1    =XCP.NFLS   (X1) = CURRENT NOMINAL FL SCM 000640  DBG=IDP    650   A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            MI     B2,EXIT.    IF ADDR BAD TO START WITH                          DBG=IDP      651     A
DBG=IDP                                                                     000660   DBG=IDP      652     A
DBG=IDP   #NFLS    ELSE                                                     000670   DBG=IDP      653     A
DBG=IDP            SA1    RA.LWP                                           000680   DBG=IDP      654     A
DBG=IDP            MI     B2,EXIT.    IF ADDR IS BAD                       000690   DBG=IDP      655     A
DBG=IDP            SX1    X1          EXTEND SIGN BIT                      000700   DBG=IDP      656     A
DBG=IDP            PL     X1,CHK2     IF CMM NOT ACTIVE                    000710   DBG=IDP      657     A
DBG=IDP            BX1    -X1         (X1) = DABA                          000720   DBG=IDP      658     A
DBG=IDP            SA1    X1          (X1) = 42/STUFF,18/FL SCM            000730   DBG=IDP      659     A
DBG=IDP            SX2    B2                                               000740   DBG=IDP      660     A
DBG=IDP            SX1    X1          (X1) = FL SCM                        000750   DBG=IDP      661     A
DBG=IDP            IX2    X2-X1       (X2) = ADDR - FL                     000760   DBG=IDP      662     A
DBG=IDP            MI     X2,EXIT.    IF ADDR LT FL                        000770   DBG=IDP      663     A
DBG=IDP   CHK2     BX2    X6          SAVE X6 ACROSS MEMORY MACRO          000780   DBG=IDP      664     A
DBG=IDP            SX6    A6                                               000790   DBG=IDP      665     A
DBG=IDP            SA6    CHKB        SAVE A6 ACROSS MEMORY MACRO          000800   DBG=IDP      666     A
DBG=IDP            MX6    0                                                000810   DBG=IDP      667     A
DBG=IDP            SA6    CHKA        CLEAR MEM REQUEST WORD               000820   DBG=IDP      668     A
DBG=IDP            MEMORY CM,A6,RCL   REQUEST CURRENT FIELD LENGTH         000830   DBG=IDP      669     A
DBG=IDP            SA1    CHKB        (X1) = SAVED A6                      000840   DBG=IDP      670     A
DBG=IDP            SA1    X1          (X1) = ((A6))                        000850   DBG=IDP      671     A
DBG=IDP            BX6    X1                                               000860   DBG=IDP      672     A
DBG=IDP            SA6    A1          RESTORE A6                           000870   DBG=IDP      673     A
DBG=IDP            BX6    X2          RESTORE X6                           000880   DBG=IDP      674     A
DBG=IDP            SA1    CHKA        (X1) = 30/FL SCM,30/STUFF            000890   DBG=IDP      675     A
DBG=IDP            AX1    30D         (X1) = FL SCM                        000900   DBG=IDP      676     A
DBG=IDP   #NFLS    ENDIF                                                   000910   DBG=IDP      677     A
DBG=IDP                                                                    000920   DBG=IDP      678     A
DBG=IDP            SX2    B2                                                          DBG=IDP      679     A
DBG=IDP            IX2    X2-X1                                                       DBG=IDP      680     A
DBG=IDP            MI     X2,EXIT.    IF ADDR OK                                  DBG=IDP      681     A
DBG=IDP            SB2    -B2         SET TO *ADDR IS BAD*                        DBG=IDP      682     A
DBG=IDP            EQ     EXIT.                                                   DBG=IDP      683     A
DBG=IDP                                                                    000940   DBG=IDP      684     A
DBG=IDP                                                                    000950   DBG=IDP      685     A
DBG=IDP    CHKA    BSSZ   1           REQUEST WORD FOR MEMORY MACRO       000960   DBG=IDP      686     A
DBG=IDP    CHKB    BSSZ   1           FOR SAVING A6 ACROSS MEMORY MACRO   000970   DBG=IDP      687     A
DBG=IDP    DAB     SPACE  4,8                                                      DBG=IDP      688     A
DBG=IDP   **       DAB - DUMP AN -A- OR -B- REGISTER AND WHAT IT POINTS TO.        DBG=IDP      689     A
DBG=IDP   *                                                                        DBG=IDP      690     A
DBG=IDP   *                                                                        DBG=IDP      691     A
DBG=IDP   *        ENTRY  (B4) = 0TR, WHERE                                        DBG=IDP      692     A
DBG=IDP   *                           T IS REGISTER TYPE                           DBG=IDP      693     A
DBG=IDP   *                                 = 0 FOR -B- REGISTER                   DBG=IDP      694     A
DBG=IDP   *                                 = 1 FOR -A- REGISTER                   DBG=IDP      695     A
DBG=IDP   *                           R IS REGISTER NR (0-7)                       DBG=IDP      696     A
DBG=IDP   *                           E.G.  A6 WOULD BE 016                        DBG=IDP      697     A
DBG=IDP   *                                                                        DBG=IDP      698     A
DBG=IDP   *        EXIT   (B4) IS UNCHANGED                                        DBG=IDP      699     A
DBG=IDP   *                                                                        DBG=IDP      700     A
DBG=IDP   *        USES   ALL BUT B4                                               DBG=IDP      701     A
DBG=IDP   *                                                                        DBG=IDP      702     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *         CALLS  FAB,PRIDP                                          DBG=IDP      703     A
DBG=IDP                                                                      DBG=IDP      704     A
DBG=IDP                                                                      DBG=IDP      705     A
DBG=IDP   DAB      SUBR                ** ENTRY/EXIT **                      DBG=IDP      706     A
DBG=IDP            SB6    SNAPLNE                                            DBG=IDP      707     A
DBG=IDP            RJ     FAB         FORMAT -A- OR -B- REGISTER             DBG=IDP      708     A
DBG=IDP            PRIDP  SNAPLNE                                            DBG=IDP      709     A
DBG=IDP            EQ     EXIT.                                              DBG=IDP      710     A
DBG=IDP   DAR      SPACE  4,8                                                DBG=IDP      711     A
DBG=IDP  **        DAR - DUMP ALL REGISTERS.                                 DBG=IDP      712     A
DBG=IDP  *                                                                   DBG=IDP      713     A
DBG=IDP  *                                                                   DBG=IDP      714     A
DBG=IDP  *         ENTRY  NONE                                               DBG=IDP      715     A
DBG=IDP  *                                                                   DBG=IDP      716     A
DBG=IDP  *         EXIT   NONE                                               DBG=IDP      717     A
DBG=IDP  *                                                                   DBG=IDP      718     A
DBG=IDP  *         USES   ALL                                                DBG=IDP      719     A
DBG=IDP  *                                                                   DBG=IDP      720     A
DBG=IDP  *         CALLS  DAB,DUX,PRIDP,WOD                                  DBG=IDP      721     A
DBG=IDP                                                                      DBG=IDP      722     A
DBG=IDP                                                                      DBG=IDP      723     A
DBG=IDP   DAR      SUBR   0            ** ENTRY/EXIT **                      DBG=IDP      724     A
DBG=IDP            SB4    B0          INITIALIZE (B4) = REGISTER COUNT, 0TR  DBG=IDP      725     A
DBG=IDP            PRIDP  ,,L.PRB                                            DBG=IDP      726     A
DBG=IDP                                                                      DBG=IDP      727     A
DBG=IDP            SA1    IDPFLG                                             DBG=IDP      728     A
DBG=IDP            LX1    59-IDF.IDOP                                        DBG=IDP      729     A
DBG=IDP            PL     X1,DAR5     IF NOT WRITING INTERACTIVELY,          DBG=IDP      730     A
DBG=IDP  *                               PACK -A- AND -B- REGISTERS...       DBG=IDP      731     A
DBG=IDP                                                                      DBG=IDP      732     A
DBG=IDP  *         DUMP -B- REGISTERS.                                       DBG=IDP      733     A
DBG=IDP                                                                      DBG=IDP      734     A
DBG=IDP   DAR2     RJ     DAB         DUMP -B- REGISTER                      DBG=IDP      735     A
DBG=IDP            SB4    B4+B1                                              DBG=IDP      736     A
DBG=IDP            SB7    007B                                               DBG=IDP      737     A
DBG=IDP            LE     B4,B7,DAR2  IF NOT DONE                           DBG=IDP      738     A
DBG=IDP                                                                      DBG=IDP      739     A
DBG=IDP            PRIDP  ,,L.PRB                                            DBG=IDP      740     A
DBG=IDP                                                                      DBG=IDP      741     A
DBG=IDP  *         DUMP -A- REGISTERS.                                       DBG=IDP      742     A
DBG=IDP                                                                      DBG=IDP      743     A
DBG=IDP   DAR3     RJ     DAB         DUMP -A- REGISTER                      DBG=IDP      744     A
DBG=IDP            SB4    B4+B1                                              DBG=IDP      745     A
DBG=IDP            SB7    017B                                               DBG=IDP      746     A
DBG=IDP            LE     B4,B7,DAR3  IF NOT DONE                           DBG=IDP      747     A
DBG=IDP                                                                      DBG=IDP      748     A
DBG=IDP            PRIDP  ,,L.PRB                                            DBG=IDP      749     A
DBG=IDP                                                                      DBG=IDP      750     A
DBG=IDP  *         DUMP -X- REGISTERS.                                       DBG=IDP      751     A
DBG=IDP                                                                      DBG=IDP      752     A
DBG=IDP   DAR4     RJ     DUX         DUMP -X- REGISTER                      DBG=IDP      753     A
DBG=IDP            SB4    B4+B1                                              DBG=IDP      754     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            SB7    027B                                                    DBG=IDP     755    A
DBG=IDP            LE     B4,B7,DAR4  IF NOT DONE                                 DBG=IDP     756    A
DBG=IDP            EQ     EXIT.                                                   DBG=IDP     757    A
DBG=IDP                                                                           DBG=IDP     758    A
DBG=IDP  *         HERE IF PACKING -A- AND -B- REGISTERS IN BATCH MODE.            DBG=IDP     759    A
DBG=IDP                                                                           DBG=IDP     760    A
DBG=IDP   DAR5     SB6    SNAPLNE                                                 DBG=IDP     761    A
DBG=IDP            RJ     FAB         FORMAT -B- REGISTER                         DBG=IDP     762    A
DBG=IDP            SB4    B4+10B                                                  DBG=IDP     763    A
DBG=IDP            SB6    A6                                                      DBG=IDP     764    A
DBG=IDP            RJ     FAB         FORMAT -A- REGISTER                         DBG=IDP     765    A
DBG=IDP            PRIDP  SNAPLNE                                                 DBG=IDP     766    A
DBG=IDP            SB4    B4-10B+1                                                DBG=IDP     767    A
DBG=IDP            SB7    007B                                                    DBG=IDP     768    A
DBG=IDP            LE     B4,B7,DAR5  IF NOT DONE                                 DBG=IDP     769    A
DBG=IDP                                                                           DBG=IDP     770    A
DBG=IDP            PRIDP  ,,L.PRB                                                 DBG=IDP     771    A
DBG=IDP            SB4    020B        (B4) = 0TR, WHERE T=2 (X REG)               DBG=IDP     772    A
DBG=IDP            EQ     DAR4                                                    DBG=IDP     773    A
DBG=IDP   DCM      SPACE  4,8                                                     DBG=IDP     774    A
DBG=IDP  **        DCM - DUMP CENTRAL MEMORY.                                     DBG=IDP     775    A
DBG=IDP  *                                                                        DBG=IDP     776    A
DBG=IDP  *                                                                        DBG=IDP     777    A
DBG=IDP  *                CONVERTS CM WORDS TO OCTAL DPC AND DUMPS THEM IN THE    DBG=IDP     778    A
DBG=IDP  *         FOLLOWING FORMAT--                                             DBG=IDP     779    A
DBG=IDP  *     COL 1         1         1         1         1                     DBG=IDP     780    A
DBG=IDP  *         .NNNNNN.+.CCCCCCCCCCCCCCCCCCCC.........ETC                000810 DBG=IDP     781    A
DBG=IDP  *                   ******************************                      DBG=IDP     782    A
DBG=IDP  *         N = ADDR OF 1ST WORD DUMPED ON LINE                            DBG=IDP     783    A
DBG=IDP  *         + = + IF *NNNNNN* IS DECK RELATIVE, ELSE                 000830 DBG=IDP     784    A
DBG=IDP  *           = BLANK(55B) IF *NNNNNN* IS ABSOLUTE ADDR.             000840 DBG=IDP     785    A
DBG=IDP  *         C = CONVERTED CM WORD                                          DBG=IDP     786    A
DBG=IDP  *         . = BLANK(55B)                                                 DBG=IDP     787    A
DBG=IDP  *         THE FORMAT OF COLS 11 THRU 40 (INDICATED BY *) IS REPEATED FOR DBG=IDP     788    A
DBG=IDP  *           HOWEVER MANY CM WORDS ARE TO BE DUMPED ON A LINE.            DBG=IDP     789    A
DBG=IDP  *                                                                        DBG=IDP     790    A
DBG=IDP  *                (IDPFLG) = 42/OTHER, 18/NR OF CM WORDS TO BE DUMPED ON  DBG=IDP     791    A
DBG=IDP  *                            A SINGLE LINE                               DBG=IDP     792    A
DBG=IDP  *                                                                        DBG=IDP     793    A
DBG=IDP  *         ENTRY  (X2) = 18/0, 21/LEV OF IND ADDR ,21/FWA                 DBG=IDP     794    A
DBG=IDP  *                (X3) = 18/0, 21/LEV OF IND ADDR ,21/LWA                 DBG=IDP     795    A
DBG=IDP  *                       (X3) IS OPTIONAL                                 DBG=IDP     796    A
DBG=IDP  *                (X4) = 18/0,21/LEV OF IND ADDR ,21/LEN                  DBG=IDP     797    A
DBG=IDP  *                       NOTE -- IF LEN = 0, DMP IS FROM FWA TO LWA,      DBG=IDP     798    A
DBG=IDP  *                                    ELSE DMP IS FROM FWA TO FWA+LEN     DBG=IDP     799    A
DBG=IDP  *                                                                        DBG=IDP     800    A
DBG=IDP  *         EXIT   (B2) = .MI. IF FWA,LWA, OR LEN BAD, ELSE .PL.           DBG=IDP     801    A
DBG=IDP  *                                                                        DBG=IDP     802    A
DBG=IDP  *         USES   ALL BUT A0  (INCLUDES ALL CALLS)                  000860 DBG=IDP     803    A
DBG=IDP  *                                                                        DBG=IDP     804    A
DBG=IDP  *         CALLS  CAD,FLL,PRIDP,WOD                                 000880 DBG=IDP     805    A
DBG=IDP                                                                           DBG=IDP     806    A
```

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP                                                                                  DBG=IDP      807     A
DBG=IDP   DCM      SUBR                  ** ENTRY/EXIT **                                DBG=IDP      808     A
DBG=IDP            RJ     FLL            CHECK FWA,LWA, AND LENGTH                       DBG=IDP      809     A
DBG=IDP            SA1    =10H                                                           DBG=IDP      810     A
DBG=IDP            MI     B2,EXIT.       IF FWA,LWA, OR LEN IS BAD                       DBG=IDP      811     A
DBG=IDP            BX6    X1                                                             DBG=IDP      812     A
DBG=IDP            SB6    B0             PRESET TO *READY FOR NEW SNAP LINE*             DBG=IDP      813     A
DBG=IDP            SA6    SNAPLNE        PRESTORE (A6) = ADDR OF LAST WORD STORED        DBG=IDP      814     A
DBG=IDP   *                                INTO SNAP LINE.  THIS WILL FORCE A BLANK      DBG=IDP      815     A
DBG=IDP   *                                LINE TO BE OUTPUT AS 1ST LINE OF DUMP.        DBG=IDP      816     A
DBG=IDP                                                                                  DBG=IDP      817     A
DBG=IDP   *        ** MAIN LOOP NODE **                                                  DBG=IDP      818     A
DBG=IDP                                                                                  DBG=IDP      819     A
DBG=IDP   DCM2     SA5    AP=FWA         (X5) = ADDR OF NEXT WORD TO DUMP                DBG=IDP      820     A
DBG=IDP            SA2    A5+B1          (X2) = LWA OF DUMP                              DBG=IDP      821     A
DBG=IDP            IX3    X2-X5                                                          DBG=IDP      822     A
DBG=IDP            SX7    X5+B1                                                          DBG=IDP      823     A
DBG=IDP            MI     X3,DCM6        IF FINISHED                                     DBG=IDP      824     A
DBG=IDP            SA7    A5                                                             DBG=IDP      825     A
DBG=IDP            GT     B6,B0,DCM3     IF MORE WORDS TO DUMP ON THIS LINE              DBG=IDP      826     A
DBG=IDP                                                                                  DBG=IDP      827     A
DBG=IDP   *        HERE IF READY TO OUTPUT OLD LINE AND START NEW LINE BY                DBG=IDP      828     A
DBG=IDP   *          SETTING UP ADDR OF 1ST WORD ON LINE.                                DBG=IDP      829     A
DBG=IDP                                                                                  DBG=IDP      830     A
DBG=IDP            MX6    0                                                              DBG=IDP      831     A
DBG=IDP            SA6    A6+B1          MARK EOL                                        DBG=IDP      832     A
DBG=IDP            PRIDP  SNAPLNE                                                        DBG=IDP      833     A
DBG=IDP                                                                                  DBG=IDP      834     A
DBG=IDP            SX1    X5                                                             DBG=IDP      835     A
DBG=IDP            RJ     CAD            CONVERT ADDR TO DPC                     000900  DBG=IDP      836     A
DBG=IDP            SA1    IDPFLG                                                         DBG=IDP      837     A
DBG=IDP            SB6    X1             (B6) = NR OF WORDS TO DUMP ON A LINE            DBG=IDP      838     A
DBG=IDP            SA6    SNAPLNE                                                        DBG=IDP      839     A
DBG=IDP            LE     B6,B0,*+4S15 IF BAD WORDS PER LINE COUNT...                    DBG=IDP      840     A
DBG=IDP                                                                                  DBG=IDP      841     A
DBG=IDP   *        HERE TO DUMP A SINGLE WORD.                                           DBG=IDP      842     A
DBG=IDP                                                                                  DBG=IDP      843     A
DBG=IDP   DCM3     SA1    X5             (X1) = NEXT WORD TO BE DUMPED                   DBG=IDP      844     A
DBG=IDP            SA2    =10H                                                           DBG=IDP      845     A
DBG=IDP            NZ     X1,DCM4        IF WORD TO DUMP IS NON-ZERO                     DBG=IDP      846     A
DBG=IDP            SX3    2R 0&2R                                                        DBG=IDP      847     A
DBG=IDP            SX4    2R-0&2R                                                        DBG=IDP      848     A
DBG=IDP            BX3    -X1*X3                                                         DBG=IDP      849     A
DBG=IDP            BX4    -X1+X4                                                         DBG=IDP      850     A
DBG=IDP            IX5    X3+X4                                                          DBG=IDP      851     A
DBG=IDP            LX6    X2                                                             DBG=IDP      852     A
DBG=IDP            BX7    X2-X5          (X7) = .........0   (.=55B)  IF (X1) = +0      DBG=IDP      853     A
DBG=IDP   *                                  = ........-0   (.=55B)  IF (X1) = -0      DBG=IDP      854     A
DBG=IDP            EQ     DCM5                                                           DBG=IDP      855     A
DBG=IDP                                                                                  DBG=IDP      856     A
DBG=IDP   DCM4     RJ     =XWOD          CONVERT A FULL BINARY WORD TO OCTAL DPC         DBG=IDP      857     A
DBG=IDP                                                                                  DBG=IDP      858     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP   DCM5     SA6    A6+B1                                                  DBG=IDP     859    A
DBG=IDP            SA7    A6+B1                                                  DBG=IDP     860    A
DBG=IDP            SA1    =10H                                                   DBG=IDP     861    A
DBG=IDP            SB6    B6-B1                                                  DBG=IDP     862    A
DBG=IDP            BX6    X1                                                     DBG=IDP     863    A
DBG=IDP            SA6    A7+B1                                                  DBG=IDP     864    A
DBG=IDP            EQ     DCM2        MAIN LOOP...                               DBG=IDP     865    A
DBG=IDP                                                                          DBG=IDP     866    A
DBG=IDP   *        HERE IF FINISHED -- NEED TO OUTPUT FINAL WORD.                DBG=IDP     867    A
DBG=IDP                                                                          DBG=IDP     868    A
DBG=IDP   DCM6     MX6    0                                                      DBG=IDP     869    A
DBG=IDP            SA6    A6+B1       MARK EOL                                   DBG=IDP     870    A
DBG=IDP            PRIDP  SNAPLNE                                                DBG=IDP     871    A
DBG=IDP            EQ     EXIT.                                                  DBG=IDP     872    A
DBG=IDP   DSR      SPACE  4,8                                                    DBG=IDP     873    A
DBG=IDP   **       DSR - DUMP SELECTED REGISTERS.                                DBG=IDP     874    A
DBG=IDP   *                                                                      DBG=IDP     875    A
DBG=IDP   *                                                                      DBG=IDP     876    A
DBG=IDP   *                PERFORMS A SELECTED REGISTER DUMP, BASED ON A PARAMETER DBG=IDP  877    A
DBG=IDP   *        LIST SET UP BY *SNAP* AND/OR *REG* MACROS.                    DBG=IDP     878    A
DBG=IDP   *                                                                      DBG=IDP     879    A
DBG=IDP   *                0TR = REGISTER DESIGNATOR                             DBG=IDP     880    A
DBG=IDP   *                           T = REGISTER TYPE                          DBG=IDP     881    A
DBG=IDP   *                             = 0 FOR -B- REGISTER                     DBG=IDP     882    A
DBG=IDP   *                             = 1 FOR -A- REGISTER                     DBG=IDP     883    A
DBG=IDP   *                             = 2 FOR -X- REGISTER                     DBG=IDP     884    A
DBG=IDP   *                           R = REGISTER NR (0-7)                      DBG=IDP     885    A
DBG=IDP   *                             E.G. A3 WOULD BE 013B                    DBG=IDP     886    A
DBG=IDP   *                SV=B = FWA OF REGISTER SAVE AREA                      DBG=IDP     887    A
DBG=IDP   *                                                                      DBG=IDP     888    A
DBG=IDP   *        PARAMETER LIST FORMAT--                                       DBG=IDP     889    A
DBG=IDP   *                                                                      DBG=IDP     890    A
DBG=IDP   *        VFD    60/SV=B+0TR                                            DBG=IDP     891    A
DBG=IDP   *        VFD    60/SV=B+0TR                                            DBG=IDP     892    A
DBG=IDP   *                .                                                     DBG=IDP     893    A
DBG=IDP   *                .                                                     DBG=IDP     894    A
DBG=IDP   *                60/0        END OF PARAMETER LIST                     DBG=IDP     895    A
DBG=IDP   *                                                                      DBG=IDP     896    A
DBG=IDP   *        ENTRY  (A1,X1) = ADDR + CNTS OF 1ST WORD OF PARAMETER LIST    DBG=IDP     897    A
DBG=IDP   *                                                                      DBG=IDP     898    A
DBG=IDP   *        EXIT   NONE                                                   DBG=IDP     899    A
DBG=IDP   *                                                                      DBG=IDP     900    A
DBG=IDP   *        USES   ALL                                                    DBG=IDP     901    A
DBG=IDP   *                                                                      DBG=IDP     902    A
DBG=IDP   *        CALLS  DAB,DUX                                                DBG=IDP     903    A
DBG=IDP                                                                          DBG=IDP     904    A
DBG=IDP                                                                          DBG=IDP     905    A
DBG=IDP   DSR      SUBR                ** ENTRY/EXIT **                          DBG=IDP     906    A
DBG=IDP                                                                          DBG=IDP     907    A
DBG=IDP   DSR2     ZR     X1,EXIT.    IF FINISHED DUMPING SELECTED REGISTERS    DBG=IDP     908    A
DBG=IDP            SB2    =XSV=B                                                 DBG=IDP     909    A
DBG=IDP            SB3    X1                                                     DBG=IDP     910    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            SX6    A1+                                                  DBG=IDP     911    A
DBG=IDP            SB4    B3-B2                                                DBG=IDP     912    A
DBG=IDP            SB5    20B                                                  DBG=IDP     913    A
DBG=IDP            SA6    DSRA        SAVE (A1) = CURRENT POSITION IN PARM LIST DBG=IDP     914    A
DBG=IDP            GE     B4,B5,DSR3  IF -X- REGISTER DUMP                     DBG=IDP     915    A
DBG=IDP            RJ     DAB         DUMP -A- OR -B- REGISTER                 DBG=IDP     916    A
DBG=IDP            EQ     DSR4                                                 DBG=IDP     917    A
DBG=IDP                                                                        DBG=IDP     918    A
DBG=IDP   DSR3     RJ     DUX         DUMP -X- REGISTER                        DBG=IDP     919    A
DBG=IDP                                                                        DBG=IDP     920    A
DBG=IDP   DSR4     SA1    DSRA                                                 DBG=IDP     921    A
DBG=IDP            SA1    X1+1                                                 DBG=IDP     922    A
DBG=IDP            EQ     DSR2                                                 DBG=IDP     923    A
DBG=IDP                                                                        DBG=IDP     924    A
DBG=IDP                                                                        DBG=IDP     925    A
DBG=IDP   DSRA     BSSZ   1           SAVED (A1) = ADDR OF NEXT REGISTER PARAME- DBG=IDP     926    A
DBG=IDP  *                              TER WORD                               DBG=IDP     927    A
DBG=IDP   DUX      SPACE  4,8                                                  DBG=IDP     928    A
DBG=IDP  **        DUX - DUMP AN -X- REGISTER.                                 DBG=IDP     929    A
DBG=IDP  *                                                                     DBG=IDP     930    A
DBG=IDP  *                                                                     DBG=IDP     931    A
DBG=IDP  *         ENTRY  (B4) = 0TR, WHERE                                    DBG=IDP     932    A
DBG=IDP  *                           T IS REGISTER TYPE (2=X REGISTER)         DBG=IDP     933    A
DBG=IDP  *                           R IS REGISTER NR (0-7)                    DBG=IDP     934    A
DBG=IDP  *                             E.G. X1 WOULD BE 021B                   DBG=IDP     935    A
DBG=IDP  *                                                                     DBG=IDP     936    A
DBG=IDP  *         EXIT   (B4) IS UNCHANGED                                    DBG=IDP     937    A
DBG=IDP  *                                                                     DBG=IDP     938    A
DBG=IDP  *         USES   ALL BUT B4                                           DBG=IDP     939    A
DBG=IDP  *                                                                     DBG=IDP     940    A
DBG=IDP  *         CALLS  PRIDP,WOD                                            DBG=IDP     941    A
DBG=IDP                                                                        DBG=IDP     942    A
DBG=IDP                                                                        DBG=IDP     943    A
DBG=IDP   DUX      SUBR               ** ENTRY/EXIT **                         DBG=IDP     944    A
DBG=IDP            SA1    B4+SV=B                                              DBG=IDP     945    A
DBG=IDP            SA2    DUXA                                                 DBG=IDP     946    A
DBG=IDP            SX7    B4-20B      (X7) = X REGISTER NR                     DBG=IDP     947    A
DBG=IDP            LX7    2*CHAR-1*CHAR                                        DBG=IDP     948    A
DBG=IDP            IX6    X2+X7       (X6) = 10H       XN   , WHERE N=0 THRU 7 DBG=IDP     949    A
DBG=IDP            SA6    SNAPLNE                                              DBG=IDP     950    A
DBG=IDP            RJ     =XWOD       CONVERT ONE FULL BINARY WORD TO OCTAL DPC DBG=IDP     951    A
DBG=IDP            SA6    A6+B1                                                DBG=IDP     952    A
DBG=IDP            SA7    A6+B1                                                DBG=IDP     953    A
DBG=IDP            SA1    =10H                                                 DBG=IDP     954    A
DBG=IDP            SA2    B4+SV=B     (X2) = CONTENTS OF X-N-                  DBG=IDP     955    A
DBG=IDP            BX6    X1                                                   DBG=IDP     956    A
DBG=IDP            LX7    X2                                                   DBG=IDP     957    A
DBG=IDP            SA6    A7+B1                                                DBG=IDP     958    A
DBG=IDP            SA7    A6+B1                                                DBG=IDP     959    A
DBG=IDP            BX6    X6-X6                                                DBG=IDP     960    A
DBG=IDP            SA6    A7+B1       MARK EOL                                 DBG=IDP     961    A
DBG=IDP            PRIDP  SNAPLNE                                              DBG=IDP     962    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            EQ     EXIT.                                                    DBG=IDP      963     A
DBG=IDP                                                                             DBG=IDP      964     A
DBG=IDP                                                                             DBG=IDP      965     A
DBG=IDP   DUXA     DATA   10H       X0                                              DBG=IDP      966     A
DBG=IDP   FAB      SPACE  4,8                                                       DBG=IDP      967     A
DBG=IDP  **        FAB - FORMAT AN -A- OR -B- REGISTER.                             DBG=IDP      968     A
DBG=IDP  *                                                                          DBG=IDP      969     A
DBG=IDP  *                                                                          DBG=IDP      970     A
DBG=IDP  *                THIS ROUTINE WILL FORMAT AN -A- OR -B- REGISTER           DBG=IDP      971     A
DBG=IDP  *         SUITABLE FOR PRINTING. FORMAT--                                  DBG=IDP      972     A
DBG=IDP  *                                                                          DBG=IDP      973     A
DBG=IDP  *    (B6)+0         0         0         0         0         0             DBG=IDP      974     A
DBG=IDP  *         .......TR..NNNNNN.....C(TR).=.NNNNNNNNNNNNNNNNNNNN                DBG=IDP      975     A
DBG=IDP  *                                                                          DBG=IDP      976     A
DBG=IDP  *         T=REGISTER TYPE (B,A),R=REGISTER NR (0-7),.=BLANK(55B)           DBG=IDP      977     A
DBG=IDP  *                                                                          DBG=IDP      978     A
DBG=IDP  *         ENTRY  (B4) = 0TR, WHERE                                         DBG=IDP      979     A
DBG=IDP  *                           T IS REGISTER TYPE (B=0,A=1)                   DBG=IDP      980     A
DBG=IDP  *                           R IS REGISTER NR (0-7)                         DBG=IDP      981     A
DBG=IDP  *                (B6) = FWA TO STORE                                       DBG=IDP      982     A
DBG=IDP  *                                                                          DBG=IDP      983     A
DBG=IDP  *         EXIT   (B4) = UNCHANGED                                          DBG=IDP      984     A
DBG=IDP  *                (B6) = UNCHANGED                                          DBG=IDP      985     A
DBG=IDP  *                (A6) = ADDR OF FULL ZERO WORD EOL MARKER                  DBG=IDP      986     A
DBG=IDP  *                                                                          DBG=IDP      987     A
DBG=IDP  *         USES   X - ALL     (INCLUDES ALL CALLS)                          DBG=IDP      988     A
DBG=IDP  *                A - ALL BUT A0                                           DBG=IDP      989     A
DBG=IDP  *                B - 2,3,4,5                                              DBG=IDP      990     A
DBG=IDP  *                                                                          DBG=IDP      991     A
DBG=IDP  *         CALLS  CHK,COD,WOD                                               DBG=IDP      992     A
DBG=IDP                                                                             DBG=IDP      993     A
DBG=IDP                                                                             DBG=IDP      994     A
DBG=IDP   FAB      SUBR               ** ENTRY/EXIT **                              DBG=IDP      995     A
DBG=IDP            SX1    B4                                                        DBG=IDP      996     A
DBG=IDP            MX0    -3                                                        DBG=IDP      997     A
DBG=IDP            SA5    SV=B+B4     (X5) = SAVED (TR)   (T=REG TYPE,R=REG NR)     DBG=IDP      998     A
DBG=IDP            BX6    X0*X1       (X6) = REG TYPE (B=00B,A=10B)                 DBG=IDP      999     A
DBG=IDP            SB5    B4          SAVE (B4) = 0TR                               DBG=IDP     1000     A
DBG=IDP            BX7    -X0*X1      (X7) = REG NR (0-7)                           DBG=IDP     1001     A
DBG=IDP            LX6    -3                                                        DBG=IDP     1002     A
DBG=IDP            SA2    FABA+X6                                                   DBG=IDP     1003     A
DBG=IDP            SA3    FABB+X6                                                   DBG=IDP     1004     A
DBG=IDP            LX7    2*CHAR-1*CHAR                                             DBG=IDP     1005     A
DBG=IDP            IX6    X2+X7       (X6) = .......TR.   (.=BLANK,T=RTYPE,R=RNR)   DBG=IDP     1006     A
DBG=IDP            LX7    5*CHAR-2*CHAR                                             DBG=IDP     1007     A
DBG=IDP            SA6    B6                                                        DBG=IDP     1008     A
DBG=IDP            MX0    -18                                                       DBG=IDP     1009     A
DBG=IDP            IX7    X3+X7       (X7) = ..C(TR).=.                             DBG=IDP     1010     A
DBG=IDP            BX1    -X0*X5                                                    DBG=IDP     1011     A
DBG=IDP            SA7    B6+2                                                      DBG=IDP     1012     A
DBG=IDP            RJ     =XCOD       CONVERT BINARY TO OCTAL DPC                   DBG=IDP     1013     A
DBG=IDP            LX6    9*CHAR-6*CHAR                                             DBG=IDP     1014     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            SB4    B5          RESTORE (B4)                                  DBG=IDP    1015    A
DBG=IDP            SA6    A6+B1                                                     DBG=IDP    1016    A
DBG=IDP            SB2    X5                                                        DBG=IDP    1017    A
DBG=IDP            RJ     CHK         CHECK CM ADDRESS                              DBG=IDP    1018    A
DBG=IDP            PL     B2,FAB2     IF ADDR OK                                    DBG=IDP    1019    A
DBG=IDP            SA1    =20H ** OUT OF RANGE **                                   DBG=IDP    1020    A
DBG=IDP            SA2    A1+B1                                                     DBG=IDP    1021    A
DBG=IDP            BX6    X1                                                        DBG=IDP    1022    A
DBG=IDP            LX7    X2                                                        DBG=IDP    1023    A
DBG=IDP            EQ     FAB3                                                      DBG=IDP    1024    A
DBG=IDP                                                                             DBG=IDP    1025    A
DBG=IDP  *         HERE TO CONVERT C(TR) TO OCTAL DPC.                               DBG=IDP    1026    A
DBG=IDP                                                                             DBG=IDP    1027    A
DBG=IDP   FAB2     SA1    B2          (X1) = C(TR)                                  DBG=IDP    1028    A
DBG=IDP            RJ     WOD         CONVERT A FULL BINARY WORD TO OCTAL DPC       DBG=IDP    1029    A
DBG=IDP                                                                             DBG=IDP    1030    A
DBG=IDP   FAB3     SA6    A7+B1                                                     DBG=IDP    1031    A
DBG=IDP            SA7    A6+B1                                                     DBG=IDP    1032    A
DBG=IDP            MX6    0                                                         DBG=IDP    1033    A
DBG=IDP            SA6    A7+B1       MARK EOL                                      DBG=IDP    1034    A
DBG=IDP            EQ     EXIT.                                                     DBG=IDP    1035    A
DBG=IDP                                                                             DBG=IDP    1036    A
DBG=IDP                                                                             DBG=IDP    1037    A
DBG=IDP   FABA     DATA   10H       B0  NNNNNN                                      DBG=IDP    1038    A
DBG=IDP            DATA   10H       A0  NNNNNN                                      DBG=IDP    1039    A
DBG=IDP   FABB     DATA   10H  C(B0) =  NNNNNNNNNNNNNNNNNNNN                        DBG=IDP    1040    A
DBG=IDP            DATA   10H  C(A0) =  NNNNNNNNNNNNNNNNNNNN                        DBG=IDP    1041    A
DBG=IDP   FLL      SPACE  4,8                                                      DBG=IDP    1042    A
DBG=IDP  **        FLL - CHECK FWA,LWA, AND LENGTH PARAMETERS.                     DBG=IDP    1043    A
DBG=IDP  *                                                                         DBG=IDP    1044    A
DBG=IDP  *                                                                         DBG=IDP    1045    A
DBG=IDP  *         ENTRY  (X2) = 18/0, 21/LEV OF IND ADDR, 21/FWA                  DBG=IDP    1046    A
DBG=IDP  *                (X3) = 18/0, 21/LEV OF IND ADDR, 21/LWA                  DBG=IDP    1047    A
DBG=IDP  *                        (X3) IS OPTIONAL                                 DBG=IDP    1048    A
DBG=IDP  *                (X4) = 18/0, 21/LEV OF IND ADDR, 21/LEN                  DBG=IDP    1049    A
DBG=IDP  *                        NOTE -- IF LEN = 0, DMP IS FROM FWA TO LWA,      DBG=IDP    1050    A
DBG=IDP  *                                     ELSE DMP IS FROM FWA TO FWA+LEN     DBG=IDP    1051    A
DBG=IDP  *                                                                         DBG=IDP    1052    A
DBG=IDP  *         EXIT   (B2) = .MI. IF FWA,LWA, OR LEN WAS BAD, ELSE .PL.        DBG=IDP    1053    A
DBG=IDP  *                (AP=FWA) = FWA                                           DBG=IDP    1054    A
DBG=IDP  *                (AP=LWA) = LWA                                           DBG=IDP    1055    A
DBG=IDP  *                                                                         DBG=IDP    1056    A
DBG=IDP  *         USES   X - ALL BUT X5                                           DBG=IDP    1057    A
DBG=IDP  *                A - 1,2,6                                                DBG=IDP    1058    A
DBG=IDP  *                B - 2,3                                                  DBG=IDP    1059    A
DBG=IDP  *                                                                         DBG=IDP    1060    A
DBG=IDP  *         CALLS  CHK,GIL                                                  DBG=IDP    1061    A
DBG=IDP                                                                            DBG=IDP    1062    A
DBG=IDP                                                                            DBG=IDP    1063    A
DBG=IDP   FLL      SUBR              ** ENTRY/EXIT **                              DBG=IDP    1064    A
DBG=IDP                                                                            DBG=IDP    1065    A
DBG=IDP  *         CHECK FWA.                                                      DBG=IDP    1066    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP                                                                                  DBG=IDP     1067     A
DBG=IDP            LX2    -21                                                            DBG=IDP     1068     A
DBG=IDP            SB3    X2+         (B3) = LEVEL OF INDIRECT ADDRESSING                DBG=IDP     1069     A
DBG=IDP            LX2    21          RESTORE (X2)                                       DBG=IDP     1070     A
DBG=IDP            RJ     GIL         GENERATE INDIRECT LOAD (IF NECESSARY)              DBG=IDP     1071     A
DBG=IDP            MI     B2,EXIT.    IF FWA IS BAD                                      DBG=IDP     1072     A
DBG=IDP            SX6    B2          SAVE (B2) = FWA                                    DBG=IDP     1073     A
DBG=IDP            SA6    AP=FWA                                                         DBG=IDP     1074     A
DBG=IDP                                                                                  DBG=IDP     1075     A
DBG=IDP  *         CHECK LENGTH.                                                        DBG=IDP     1076     A
DBG=IDP                                                                                  DBG=IDP     1077     A
DBG=IDP   FLL2     MX0    -42                                                            DBG=IDP     1078     A
DBG=IDP            BX2    -X0*X4                                                         DBG=IDP     1079     A
DBG=IDP            ZR     X2,FLL3     IF NO LENGTH, USE LWA                              DBG=IDP     1080     A
DBG=IDP            LX2    -21                                                            DBG=IDP     1081     A
DBG=IDP            SB3    X2                                                             DBG=IDP     1082     A
DBG=IDP            LX2    21                                                             DBG=IDP     1083     A
DBG=IDP            RJ     GIL         GENERATE INDIRECT LOAD (IF NECESSARY)              DBG=IDP     1084     A
DBG=IDP            MI     B2,EXIT.    IF LENGTH IS BAD                                   DBG=IDP     1085     A
DBG=IDP            SB2    X6+B2       (B2) = FWA + LEN                                   DBG=IDP     1086     A
DBG=IDP            SB2    B2-B1       (B2) = LWA                                         DBG=IDP     1087     A
DBG=IDP            RJ     CHK         CHECK CM ADDR                                      DBG=IDP     1088     A
DBG=IDP            MI     B2,EXIT.    IF LWA IS BAD                                      DBG=IDP     1089     A
DBG=IDP            SX6    B2          SAVE (B2) = LWA                                    DBG=IDP     1090     A
DBG=IDP            SA6    A6+B1                                                          DBG=IDP     1091     A
DBG=IDP            EQ     EXIT.                                                          DBG=IDP     1092     A
DBG=IDP                                                                                  DBG=IDP     1093     A
DBG=IDP  *         CHECK LWA.                                                           DBG=IDP     1094     A
DBG=IDP                                                                                  DBG=IDP     1095     A
DBG=IDP   FLL3     BX2    X3                                                             DBG=IDP     1096     A
DBG=IDP            LX2    -21                                                            DBG=IDP     1097     A
DBG=IDP            BX7    -X0*X3                                                         DBG=IDP     1098     A
DBG=IDP            SB3    X2                                                             DBG=IDP     1099     A
DBG=IDP            SX4    B1          (X4) = DEFAULT LEN IS 1, IF LEN AND LWA =0         DBG=IDP     1100     A
DBG=IDP            LX2    21                                                             DBG=IDP     1101     A
DBG=IDP            ZR     X7,FLL2     IF NO LWA, DEFAULT LEN=1                           DBG=IDP     1102     A
DBG=IDP            RJ     GIL         GENERATE INDIRECT LOAD (IF NECESSARY)              DBG=IDP     1103     A
DBG=IDP            MI     B2,EXIT.    IF LWA IS BAD                                      DBG=IDP     1104     A
DBG=IDP            SX6    B2          SAVE (B2) = LWA                                    DBG=IDP     1105     A
DBG=IDP            SA6    A6+B1                                                          DBG=IDP     1106     A
DBG=IDP            EQ     EXIT.                                                          DBG=IDP     1107     A
DBG=IDP   FRK      SPACE  4,8                                                            DBG=IDP     1108     A
DBG=IDP  **        FRK - CHECK FREQUENCY PARAMETERS.                                    DBG=IDP     1109     A
DBG=IDP  *                                                                               DBG=IDP     1110     A
DBG=IDP  *                                                                               DBG=IDP     1111     A
DBG=IDP  *         ENTRY  (X5)   =  FWA OF SNAP PARAMETER LIST                          DBG=IDP     1112     A
DBG=IDP  *                                                                               DBG=IDP     1113     A
DBG=IDP  *         FREQUENCY PARAMETER LIST AT (X5)+SN=FRK --                           DBG=IDP     1114     A
DBG=IDP  *                                                                               DBG=IDP     1115     A
DBG=IDP  *         VFD    60/LOWER LIMIT (LL)                                           DBG=IDP     1116     A
DBG=IDP  *         VFD    60/UPPER LIMIT (UL)                                           DBG=IDP     1117     A
DBG=IDP  *         VFD    60/INCREMENT  (INC)                                           DBG=IDP     1118     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *         VFD    60/10HNAME                                          DBG=IDP    1119    A
DBG=IDP  *         VFD    60/SNAP COUNT                                       DBG=IDP    1120    A
DBG=IDP  *                                                                    DBG=IDP    1121    A
DBG=IDP  *         EXIT   (X5)   =  .ZR. IF NO SNAP THIS TIME, ELSE UNCHANGED DBG=IDP    1122    A
DBG=IDP  *                                                                    DBG=IDP    1123    A
DBG=IDP  *         USES   X - ALL     (BUT X5 IF SNAP THIS TIME)             DBG=IDP    1124    A
DBG=IDP  *                A - 1,2,3,4                                        DBG=IDP    1125    A
DBG=IDP  *                B - 7                                              DBG=IDP    1126    A
DBG=IDP  *                                                                    DBG=IDP    1127    A
DBG=IDP  *         CALLS  NONE                                               DBG=IDP    1128    A
DBG=IDP                                                                       DBG=IDP    1129    A
DBG=IDP                                                                       DBG=IDP    1130    A
DBG=IDP   FRK.NO   SX5    0                                                   DBG=IDP    1131    A
DBG=IDP                                                                       DBG=IDP    1132    A
DBG=IDP   FRK      SUBR               ** ENTRY/EXIT **                        DBG=IDP    1133    A
DBG=IDP            SA1    X5+SN=LL    (X1) = LL                               DBG=IDP    1134    A
DBG=IDP            SA2    A1+B1       (X2) = UL                               DBG=IDP    1135    A
DBG=IDP            SA3    A2+B1       (X3) = INC                              DBG=IDP    1136    A
DBG=IDP            SA4    X5+SN=CNT   (X4) = SNAP COUNT                       DBG=IDP    1137    A
DBG=IDP            MI     X1,FRK.NO   IF LL BAD                               DBG=IDP    1138    A
DBG=IDP            MI     X2,FRK.NO   IF UL BAD                               DBG=IDP    1139    A
DBG=IDP            MI     X3,FRK.NO    IF INC BAD                             DBG=IDP    1140    A
DBG=IDP            SX6    X4+B1       (X6) = SNAP COUNT + 1                   DBG=IDP    1141    A
DBG=IDP            SA6    A4+                                                 DBG=IDP    1142    A
DBG=IDP            IX7    X6-X1       (X7) = SNAP CNT - LOWER LIMIT           DBG=IDP    1143    A
DBG=IDP            MI     X7,FRK.NO   IF NOT TIME TO SNAP YET                 DBG=IDP    1144    A
DBG=IDP            IX7    X2-X6       (X7) = UPPER LIMIT - SNAP COUNT         DBG=IDP    1145    A
DBG=IDP            PX3    X3                                                  DBG=IDP    1146    A
DBG=IDP            MI     X7,FRK.NO   IF PAST UPPER LIMIT                     DBG=IDP    1147    A
DBG=IDP                                                                       DBG=IDP    1148    A
DBG=IDP  *         COMPUTE  (COUNT/INC)*INC-COUNT.                           DBG=IDP    1149    A
DBG=IDP                                                                       DBG=IDP    1150    A
DBG=IDP            PX6    X6                                                  DBG=IDP    1151    A
DBG=IDP            NX3    X3                                                  DBG=IDP    1152    A
DBG=IDP            NX6    X6                                                  DBG=IDP    1153    A
DBG=IDP            FX7    X6/X3                                               DBG=IDP    1154    A
DBG=IDP            UX7    X7,B7                                               DBG=IDP    1155    A
DBG=IDP            LX7    X7,B7                                               DBG=IDP    1156    A
DBG=IDP            PX7    X7                                                  DBG=IDP    1157    A
DBG=IDP            NX7    X7                                                  DBG=IDP    1158    A
DBG=IDP            FX7    X7*X3                                               DBG=IDP    1159    A
DBG=IDP            IX7    X7-X6                                               DBG=IDP    1160    A
DBG=IDP            NZ     X7,FRK.NO   IF NOT AT A SNAP INCREMENT              DBG=IDP    1161    A
DBG=IDP            EQ     EXIT.       SNAP SHOULD BE HONORED...               DBG=IDP    1162    A
DBG=IDP   GIL      SPACE  4,8                                                 DBG=IDP    1163    A
DBG=IDP  **        GIL - GENERATE INDIRECT LOAD.                             DBG=IDP    1164    A
DBG=IDP  *                                                                    DBG=IDP    1165    A
DBG=IDP  *                                                                    DBG=IDP    1166    A
DBG=IDP  *                GENERATES 0 THRU N INDIRECT LOADS, CHECKING THE ADDRESS DBG=IDP 1167 A
DBG=IDP  *         TO BE LOADED BEFORE EACH LOAD.                            DBG=IDP    1168    A
DBG=IDP  *                                                                    DBG=IDP    1169    A
DBG=IDP  *         ENTRY  (X2) = ADDRESS (BITS 59-18) IGNORED)               DBG=IDP    1170    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *                (B3) = LEVEL OF INDIRECT ADDRESSING,                  DBG=IDP   1171   A
DBG=IDP  *                        (B3) = 0 MEANS DIRECT ADDRESSING              DBG=IDP   1172   A
DBG=IDP  *                                                                      DBG=IDP   1173   A
DBG=IDP  *         EXIT   (B2) = ADDRESS AT END OF INDIRECT CHAIN, IF (B2) .PL. DBG=IDP   1174   A
DBG=IDP  *                      IF (B2) .MI., ONE OF THE ADDRESSES WAS BAD      DBG=IDP   1175   A
DBG=IDP  *                                                                      DBG=IDP   1176   A
DBG=IDP  *         USES   X - 1,2                                               DBG=IDP   1177   A
DBG=IDP  *                A - 1,2                                               DBG=IDP   1178   A
DBG=IDP  *                B - 2,3                                               DBG=IDP   1179   A
DBG=IDP  *                                                                      DBG=IDP   1180   A
DBG=IDP  *         CALLS  CHK                                                   DBG=IDP   1181   A
DBG=IDP                                                                         DBG=IDP   1182   A
DBG=IDP                                                                         DBG=IDP   1183   A
DBG=IDP   GIL      SUBR              ** ENTRY/EXIT **                           DBG=IDP   1184   A
DBG=IDP                                                                         DBG=IDP   1185   A
DBG=IDP   GIL2     SB2    X2                                                    DBG=IDP   1186   A
DBG=IDP            SB3    B3-B1                                                 DBG=IDP   1187   A
DBG=IDP            RJ     CHK         CHECK CM ADDRESS                          DBG=IDP   1188   A
DBG=IDP            MI     B2,EXIT.    IF ADDR IS BAD                            DBG=IDP   1189   A
DBG=IDP            SA2    B2+                                                   DBG=IDP   1190   A
DBG=IDP            GE     B3,B0,GIL2  IF MORE LEVELS OF INDIRECT ADDRESSING TO GO DBG=IDP 1191   A
DBG=IDP            EQ     EXIT.                                                 DBG=IDP   1192   A
DBG=IDP   HDR      SPACE  4,8                                                   DBG=IDP   1193   A
DBG=IDP  **        HDR - PRINT SNAP HEADER.                                     DBG=IDP   1194   A
DBG=IDP  *                                                                      DBG=IDP   1195   A
DBG=IDP  *                                                                      DBG=IDP   1196   A
DBG=IDP  *                PRINTS A SNAP HEADER OF THE FORM--                    DBG=IDP   1197   A
DBG=IDP  *                                                                      DBG=IDP   1198   A
DBG=IDP  * NAME      SNAP NR  NNNNNN    CALLED BY NNNNNN IN XXXXXXX              DBG=IDP   1199   A
DBG=IDP  *                                                                      DBG=IDP   1200   A
DBG=IDP  *         ENTRY  (A5,    =  ADDR OF CALLER                             DBG=IDP   1201   A
DBG=IDP  *                    X5)=  FWA OF PARAMETER LSIT                       DBG=IDP   1202   A
DBG=IDP  *                                                                      DBG=IDP   1203   A
DBG=IDP  *         EXIT   NONE                                                  DBG=IDP   1204   A
DBG=IDP  *                                                                      DBG=IDP   1205   A
DBG=IDP  *         USES   ALL BUT A5,X5                                         DBG=IDP   1206   A
DBG=IDP  *                                                                      DBG=IDP   1207   A
DBG=IDP  *         CALLS  CDD,FRA=,PRIDP                                        DBG=IDP   1208   A
DBG=IDP                                                                         DBG=IDP   1209   A
DBG=IDP                                                                         DBG=IDP   1210   A
DBG=IDP   HDR      SUBR              ** ENTRY/EXIT **                           DBG=IDP   1211   A
DBG=IDP            SA1    X5+SN=HDR                                             DBG=IDP   1212   A
DBG=IDP            SA4    X5+SN=CNT                                             DBG=IDP   1213   A
DBG=IDP            ZR     X1,EXIT.    IF NO HEADER TO BE GENERATED              DBG=IDP   1214   A
DBG=IDP            MX0    8*CHAR                                                DBG=IDP   1215   A
DBG=IDP            BX6    X0*X1                                                 DBG=IDP   1216   A
DBG=IDP            SX7    2R                                                    DBG=IDP   1217   A
DBG=IDP            BX6    X6+X7                                                 DBG=IDP   1218   A
DBG=IDP            LX6    -2*CHAR     (X6) = 10H  NAME                          DBG=IDP   1219   A
DBG=IDP            SA6    SNAPLNE                                               DBG=IDP   1220   A
DBG=IDP            SA2    HDRA                                                  DBG=IDP   1221   A
DBG=IDP            SA3    A2+B1                                                 DBG=IDP   1222   A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            SX1    X4                                                    DBG=IDP    1223    A
DBG=IDP            BX6    X2                                                    DBG=IDP    1224    A
DBG=IDP            LX7    X3                                                    DBG=IDP    1225    A
DBG=IDP            SA6    A6+B1                                                 DBG=IDP    1226    A
DBG=IDP            SA7    A6+2                                                  DBG=IDP    1227    A
DBG=IDP            RJ     =XCDD       CONVERT BINARY TO DECIMAL DPC             DBG=IDP    1228    A
DBG=IDP            BX6    X4                                                    DBG=IDP    1229    A
DBG=IDP            SA6    A6+B1                                                 DBG=IDP    1230    A
DBG=IDP            SX1    A5                                                    DBG=IDP    1231    A
DBG=IDP                                                                         DBG=IDP    1232    A
DBG=IDP   #FRA     IF     DEF,FRA=                                              DBG=IDP    1233    A
DBG=IDP            SB7    HDR2        (B7) = RETURN ADDR FOR *FRA=*             DBG=IDP    1234    A
DBG=IDP            EQ     =XFRA=      FIND RELATIVE ADDRESS                     DBG=IDP    1235    A
DBG=IDP                                                                         DBG=IDP    1236    A
DBG=IDP   HDR2     BSS    0                                                     DBG=IDP    1237    A
DBG=IDP                                                                         DBG=IDP    1238    A
DBG=IDP   #FRA     ELSE                                                         DBG=IDP    1239    A
DBG=IDP            RJ     =XCOD       CONVERT BINARY TO OCTAL DPC               DBG=IDP    1240    A
DBG=IDP            BX6    X4                                                    DBG=IDP    1241    A
DBG=IDP            MX7    0                                                     DBG=IDP    1242    A
DBG=IDP   #FRA     ENDIF                                                        DBG=IDP    1243    A
DBG=IDP                                                                         DBG=IDP    1244    A
DBG=IDP            SA6    A7+B1                                                 DBG=IDP    1245    A
DBG=IDP            SA7    A6+B1                                                 DBG=IDP    1246    A
DBG=IDP            PRIDP  SNAPLNE,,L.PRB                                        DBG=IDP    1247    A
DBG=IDP            EQ     EXIT.                                                 DBG=IDP    1248    A
DBG=IDP                                                                         DBG=IDP    1249    A
DBG=IDP                                                                         DBG=IDP    1250    A
DBG=IDP  *         DATA   10H  SNAPNAME                                         DBG=IDP    1251    A
DBG=IDP   HDRA     DATA   10H SNAP NR                                           DBG=IDP    1252    A
DBG=IDP  *         DATA   10HNNNNNN                                             DBG=IDP    1253    A
DBG=IDP            DATA   10HCALLED BY                                          DBG=IDP    1254    A
DBG=IDP  *         DATA   10HNNNNNN IN                                          DBG=IDP    1255    A
DBG=IDP  *         DATA   10CXXXXXXX                                            DBG=IDP    1256    A
DBG=IDP   REG      SPACE  4,8                                                   DBG=IDP    1257    A
DBG=IDP  **        REG - REGISTER SNAPSHOT.                                     DBG=IDP    1258    A
DBG=IDP  *                                                                      DBG=IDP    1259    A
DBG=IDP  *                                                                      DBG=IDP    1260    A
DBG=IDP  *                CALLED BY *REG* MACRO.                                DBG=IDP    1261    A
DBG=IDP  *                                                                      DBG=IDP    1262    A
DBG=IDP  *         ENTRY  LOWER HALF OF *RJ REG=* WORD HAS FWA OF PARAMETER LIST.  DBG=IDP    1263    A
DBG=IDP  *                                                                      DBG=IDP    1264    A
DBG=IDP  * +       RJ     REG=                                                  DBG=IDP    1265    A
DBG=IDP  * -       VFD    30/FWA OF PARAMETER LIST                              DBG=IDP    1266    A
DBG=IDP  *                                                                      DBG=IDP    1267    A
DBG=IDP  *         PARAMETER LIST EXISTS IN LOCAL BLOCK *USE DEBUG*, AS SET UP BY  DBG=IDP    1268    A
DBG=IDP  *         *REG* MACRO (SEE *DBG=MAC*) --                               DBG=IDP    1269    A
DBG=IDP  *                                                                      DBG=IDP    1270    A
DBG=IDP  *         VFD    60/LOWER LIMIT (LL)                                   DBG=IDP    1271    A
DBG=IDP  *         VFD    60/UPPER LIMIT (UL)                                   DBG=IDP    1272    A
DBG=IDP  *         VFD    60/INCREMENT (INC)                                    DBG=IDP    1273    A
DBG=IDP  *         VFD    60/10HNAME  (OR .ZR. IF NO NAME)                      DBG=IDP    1274    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *        VFD    60/0        (USED BY *FRK* TO KEEP SNAP COUNT)                DBG=IDP   1275   A
DBG=IDP  *        VFD    60/USER FLAGS                                        001060   DBG=IDP   1276   A
DBG=IDP  *        VFD    60/REGISTER DUMP FLAG                                         DBG=IDP   1277   A
DBG=IDP  *               .                                                             DBG=IDP   1278   A
DBG=IDP  *               .                                                             DBG=IDP   1279   A
DBG=IDP  *        VFD    60/REGISTER DUMP FLAG                                         DBG=IDP   1280   A
DBG=IDP  *               .ZR. = NO REGISTER DUMP  (OR END OF REGISTER LIST)            DBG=IDP   1281   A
DBG=IDP  *               .MI. = DUMP ALL REGISTERS                                     DBG=IDP   1282   A
DBG=IDP  *               .GT.0 = ADDR OF REGISTER SAVE WORD FOR REGISTER TO DMP        DBG=IDP   1283   A
DBG=IDP  *                                                                             DBG=IDP   1284   A
DBG=IDP  *        EXIT   NONE                                                           DBG=IDP   1285   A
DBG=IDP  *                                                                             DBG=IDP   1286   A
DBG=IDP  *        USES   NONE        (ALL REGISTERS ARE SAVED AND RESTORED)             DBG=IDP   1287   A
DBG=IDP  *                                                                             DBG=IDP   1288   A
DBG=IDP  *        CALLS  DAR,DSR,FRK,HDR,RSR=,SOB,SVR=,URO=(IF DEF)                     DBG=IDP   1289   A
DBG=IDP                                                                                DBG=IDP   1290   A
DBG=IDP                                                                                DBG=IDP   1291   A
DBG=IDP   REG=    SUBR               ** ENTRY/EXIT **                                  DBG=IDP   1292   A
DBG=IDP           RJ     =XSVR=      SAVE ALL REGISTERS                                DBG=IDP   1293   A
DBG=IDP           SX1    1           SET TO *EXECUTIVE IS /REG=/*                      DBG=IDP   1294   A
DBG=IDP           RJ     SOB         SET OUTPUT BIT FLAGS                              DBG=IDP   1295   A
DBG=IDP           SA4    REG=        (X4) = 30/EQ CALLING ADDRESS+1, 30/0              DBG=IDP   1296   A
DBG=IDP           LX4    30                                                            DBG=IDP   1297   A
DBG=IDP           SA5    X4-1        (A5,X5) = ADDR + CNTS OF CALLING *RJ*             DBG=IDP   1298   A
DBG=IDP           SX6    X5          (X6) = FWA OF PARAMETER LIST                      DBG=IDP   1299   A
DBG=IDP           SA6    FWAPARM                                                       DBG=IDP   1300   A
DBG=IDP                                                                      001110   DBG=IDP   1301   A
DBG=IDP   #URO    IF     DEF,URO=                                             001120   DBG=IDP   1302   A
DBG=IDP           RJ     =XURO=      USER REG= OWNCODE                        001130   DBG=IDP   1303   A
DBG=IDP           PL     X1,REG3     IF NO REGISTER SNAPSHOT THIS TIME        001140   DBG=IDP   1304   A
DBG=IDP   #URO    ENDIF                                                        001150   DBG=IDP   1305   A
DBG=IDP                                                                      001160   DBG=IDP   1306   A
DBG=IDP           SA5    FWAPARM     (X5) = FWA OF PARAMETER LIST             001165   DBG=IDP   1307   A
DBG=IDP           RJ     FRK         CHECK FREQUENCY PARAMETERS                        DBG=IDP   1308   A
DBG=IDP           ZR     X5,REG3     IF NOT SNAPPING THIS TIME                         DBG=IDP   1309   A
DBG=IDP           RJ     HDR         OUTPUT SNAP HEADER                                DBG=IDP   1310   A
DBG=IDP           SA5    FWAPARM                                                       DBG=IDP   1311   A
DBG=IDP           SA1    X5+SN=RRL   (A1,X1) = ADDR + CNTS OF 1ST REGISTER DUMP        DBG=IDP   1312   A
DBG=IDP  *                              PARAMETER WORD                                 DBG=IDP   1313   A
DBG=IDP           PL     X1,REG2     IF NOT DUMPING ALL REGISTERS                      DBG=IDP   1314   A
DBG=IDP           RJ     DAR         DUMP ALL REGISTERS                                DBG=IDP   1315   A
DBG=IDP           EQ     REG3                                                          DBG=IDP   1316   A
DBG=IDP                                                                                DBG=IDP   1317   A
DBG=IDP   REG2    RJ     DSR         DUMP SELECTED REGISTERS                           DBG=IDP   1318   A
DBG=IDP                                                                                DBG=IDP   1319   A
DBG=IDP   REG3    RJ     =XRSR=      RESTORE ALL REGISTERS                             DBG=IDP   1320   A
DBG=IDP           EQ     EXIT.                                                         DBG=IDP   1321   A
DBG=IDP   ROL     SPACE  4,8                                                           DBG=IDP   1322   A
DBG=IDP  **       ROL - WRITE OUTPUT LINE.                                             DBG=IDP   1323   A
DBG=IDP  *                                                                             DBG=IDP   1324   A
DBG=IDP  *                                                                             DBG=IDP   1325   A
DBG=IDP  *                SHOULD BE ACCESSED VIA *PRIDP* MACRO.  THIS ROUTINE          DBG=IDP   1326   A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *         WILL OUTPUT A SINGLE LINE WITH A SPECIFIED NUMBER OF PRECEDING          DBG=IDP    1327    A
DBG=IDP  *         BLANK LINES TO EITHER THE INTERACTIVE DEBUG OUTPUT FILE                 DBG=IDP    1328    A
DBG=IDP  *         *F.IDO* AND/OR TO THE USERS BATCH DEBUG OUTPUT FILE                     DBG=IDP    1329    A
DBG=IDP  *         (VIA *PRBDO* MACRO).                                                    DBG=IDP    1330    A
DBG=IDP  *                                                                                 DBG=IDP    1331    A
DBG=IDP  *         ENTRY  (X4) = NR OF PRECEDING BLANK LINES TO OUTPUT (0=NONE)            DBG=IDP    1332    A
DBG=IDP  *                (X6) = FWA OF LINE TO BE OUTPUT                                  DBG=IDP    1333    A
DBG=IDP  *                (X7) = LEN OF LINE TO BE OUTPUT (IN WORDS)                       DBG=IDP    1334    A
DBG=IDP  *                     = .ZR. IF LEN TO BE COMPUTED BY *WRITEC* (SCOPE 2)          DBG=IDP    1335    A
DBG=IDP  *                (IDPFLG) --                                                      DBG=IDP    1336    A
DBG=IDP  *                  BIT IDF.IDOP = 1 IF WRITING TO *F.IDO*, ELSE 0                 DBG=IDP    1337    A
DBG=IDP  *                  BIT IDF.BDOP = 1 IF WRITING TO *F.BDO*, ELSE 0                 DBG=IDP    1338    A
DBG=IDP  *                                                                                 DBG=IDP    1339    A
DBG=IDP  *         EXIT   NONE                                                             DBG=IDP    1340    A
DBG=IDP  *                                                                                 DBG=IDP    1341    A
DBG=IDP  *         USES   ALL BUT B4,A0,X0,A5,X5          (INCLUDES ALL CALLS)             DBG=IDP    1342    A
DBG=IDP  *                                                                                 DBG=IDP    1343    A
DBG=IDP  *         CALLS  PRBDO,WRITEC                                                     DBG=IDP    1344    A
DBG=IDP                                                                                    DBG=IDP    1345    A
DBG=IDP                                                                                    DBG=IDP    1346    A
DBG=IDP   ROL      SUBR               ** ENTRY/EXIT **                                     DBG=IDP    1347    A
DBG=IDP                                                                                    DBG=IDP    1348    A
DBG=IDP  *         INITIALIZATION.  SAVE CRITICAL REGISTERS TO AVOID LETTING     000280   DBG=IDP    1349    A
DBG=IDP  *         HOST DESTROY THEM VIA *PRBDO* CALL.                           000290   DBG=IDP    1350    A
DBG=IDP                                                                                    DBG=IDP    1351    A
DBG=IDP            SA6    ROLA        SAVE (X6) = FWA OF LINE                              DBG=IDP    1352    A
DBG=IDP            SA7    A6+B1       SAVE (X7) = LEN OF LINE                              DBG=IDP    1353    A
DBG=IDP            BX6    X4                                                               DBG=IDP    1354    A
DBG=IDP            SX7    B4-B0                                                            DBG=IDP    1355    A
DBG=IDP            SA6    A7+B1       SAVE (X4) = NR OF PRECEDING BLANK LINES              DBG=IDP    1356    A
DBG=IDP            SA7    A6+B1       SAVE (B4)                                            DBG=IDP    1357    A
DBG=IDP            SX6    A0-B0                                                   000310   DBG=IDP    1358    A
DBG=IDP            SX7    A5-B0                                                   000320   DBG=IDP    1359    A
DBG=IDP            SA6    A7+B1       SAVE (A0)                                   000330   DBG=IDP    1360    A
DBG=IDP            SA7    A6+B1       SAVE (A5)                                   000340   DBG=IDP    1361    A
DBG=IDP            BX6    X0                                                      000350   DBG=IDP    1362    A
DBG=IDP            LX7    X5                                                      000360   DBG=IDP    1363    A
DBG=IDP            SA6    A7+B1       SAVE (X0)                                   000370   DBG=IDP    1364    A
DBG=IDP            SA7    A6+B1       SAVE (X5)                                   000380   DBG=IDP    1365    A
DBG=IDP                                                                                    DBG=IDP    1366    A
DBG=IDP  *         HERE TO OUTPUT APPROPRIATE NUMBER OF PRECEDING BLANK LINES.             DBG=IDP    1367    A
DBG=IDP                                                                                    DBG=IDP    1368    A
DBG=IDP   ROL2     SA1    ROL=PRB     (X1) = NR OF PRECEDING BLANK LINES                   DBG=IDP    1369    A
DBG=IDP            SA2    IDPFLG                                                           DBG=IDP    1370    A
DBG=IDP            SX6    X1-1                                                             DBG=IDP    1371    A
DBG=IDP            MI     X6,ROL4     IF NO MORE BLANK LINES                               DBG=IDP    1372    A
DBG=IDP            LX2    59-IDF.IDOP                                                      DBG=IDP    1373    A
DBG=IDP            SA6    A1                                                               DBG=IDP    1374    A
DBG=IDP            PL     X2,ROL3     IF NOT WRITING TO INTERACTIVE DEBUG OUTPUT           DBG=IDP    1375    A
DBG=IDP                                                                                    DBG=IDP    1376    A
DBG=IDP   #OS2     IFNE   .OS,2       IF NOT SCOPE 2                                       DBG=IDP    1377    A
DBG=IDP            WRITEC =XF.IDO,(=C= =),1                                                DBG=IDP    1378    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP                                                                                  DBG=IDP     1379     A
DBG=IDP   #OS2     ELSE                                                                  DBG=IDP     1380     A
DBG=IDP            EQ     *+4S15      IF TRYING TO WRITE INTERACTIVELY...                DBG=IDP     1381     A
DBG=IDP   #OS2     ENDIF                                                                 DBG=IDP     1382     A
DBG=IDP                                                                                  DBG=IDP     1383     A
DBG=IDP   ROL3     SA1    IDPFLG                                                         DBG=IDP     1384     A
DBG=IDP            LX1    59-IDF.BDOP                                                    DBG=IDP     1385     A
DBG=IDP            PL     X1,ROL2     IF NOT WRITING TO BATCH DEBUG OUTPUT               DBG=IDP     1386     A
DBG=IDP            PRBDO  (=C= =),1                                                      DBG=IDP     1387     A
DBG=IDP            EQ     ROL2                                                           DBG=IDP     1388     A
DBG=IDP                                                                                  DBG=IDP     1389     A
DBG=IDP   *        HERE TO OUTPUT LINE FINALLY.                                          DBG=IDP     1390     A
DBG=IDP                                                                                  DBG=IDP     1391     A
DBG=IDP   ROL4     SA1    IDPFLG                                                         DBG=IDP     1392     A
DBG=IDP            LX1    59-IDF.IDOP                                                    DBG=IDP     1393     A
DBG=IDP            PL     X1,ROL5     IF NOT WRITING TO INTERACTIVE DEBUG OUTPUT         DBG=IDP     1394     A
DBG=IDP                                                                                  DBG=IDP     1395     A
DBG=IDP   #OS2     IFNE   .OS,2       IF NOT SCOPE 2                                     DBG=IDP     1396     A
DBG=IDP            SA1    ROLA        (X1) = FWA OF LINE                                 DBG=IDP     1397     A
DBG=IDP            SA2    A1+B1       (X2) = LEN OF LINE                                 DBG=IDP     1398     A
DBG=IDP            ZR     X1,ROL6     IF NO LINE TO OUTPUT                               DBG=IDP     1399     A
DBG=IDP            WRITEC =XF.IDO,X1,X2                                                  DBG=IDP     1400     A
DBG=IDP                                                                                  DBG=IDP     1401     A
DBG=IDP   #OS2     ELSE                                                                  DBG=IDP     1402     A
DBG=IDP            EQ     *+4S15      IF TRYING TO WRITE INTERACTIVELY...                DBG=IDP     1403     A
DBG=IDP   #OS2     ENDIF                                                                 DBG=IDP     1404     A
DBG=IDP                                                                                  DBG=IDP     1405     A
DBG=IDP   ROL5     SA1    IDPFLG                                                         DBG=IDP     1406     A
DBG=IDP            SA2    ROLA                                                           DBG=IDP     1407     A
DBG=IDP            SA3    A2+B1                                                          DBG=IDP     1408     A
DBG=IDP            LX1    59-IDF.BDOP                                                    DBG=IDP     1409     A
DBG=IDP            PL     X1,ROL6     IF NOT WRITING TO BATCH DEBUG OUTPUT               DBG=IDP     1410     A
DBG=IDP            ZR     X2,ROL6     IF NO LINE TO OUTPUT                               DBG=IDP     1411     A
DBG=IDP            PRBDO  X2,X3                                                          DBG=IDP     1412     A
DBG=IDP                                                                                  DBG=IDP     1413     A
DBG=IDP   *        FINAL PROCESSING -- RESTORE SAVED REGISTERS.                          DBG=IDP     1414     A
DBG=IDP                                                                                  DBG=IDP     1415     A
DBG=IDP   ROL6     SA1    ROL=B4                                                         DBG=IDP     1416     A
DBG=IDP            SA2    A1+B1                                                  000400  DBG=IDP     1417     A
DBG=IDP            SA3    A2+B1                                                  000410  DBG=IDP     1418     A
DBG=IDP            SA4    A3+B1                                                  000420  DBG=IDP     1419     A
DBG=IDP            SB4    X1          RESTORE (B4)                               000430  DBG=IDP     1420     A
DBG=IDP            SA0    X2          RESTORE (A0)                               000440  DBG=IDP     1421     A
DBG=IDP            SA5    X3          RESTORE (A5)                               000450  DBG=IDP     1422     A
DBG=IDP            SA1    A4+B1                                                  000460  DBG=IDP     1423     A
DBG=IDP            NO                                                            000470  DBG=IDP     1424     A
DBG=IDP            BX0    X4          RESTORE (X0)                               000480  DBG=IDP     1425     A
DBG=IDP            LX5    X1          RESTORE (X5)                               000490  DBG=IDP     1426     A
DBG=IDP            EQ     EXIT.                                                          DBG=IDP     1427     A
DBG=IDP                                                                                  DBG=IDP     1428     A
DBG=IDP                                                                                  DBG=IDP     1429     A
DBG=IDP   ROLA     BSSZ   8                                                      000510  DBG=IDP     1430     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP                                                                                  DBG=IDP     1431     A
DBG=IDP            LOC    ROLA                                                           DBG=IDP     1432     A
DBG=IDP   ROL=FWA  =      *           SAVED FWA                                          DBG=IDP     1433     A
DBG=IDP   ROL=LEN  =      *+1         SAVED LENGTH                                       DBG=IDP     1434     A
DBG=IDP   ROL=PRB  =      *+2         SAVED PRECEDING BLANK COUNT                        DBG=IDP     1435     A
DBG=IDP   ROL=B4   =      *+3         SAVED (B4)                                         DBG=IDP     1436     A
DBG=IDP   ROL=A0   =      *+4         SAVED (A0)                                 000530  DBG=IDP     1437     A
DBG=IDP   ROL=A5   =      *+5         SAVED (A5)                                 000540  DBG=IDP     1438     A
DBG=IDP   ROL=X0   =      *+6         SAVED (X0)                                 000550  DBG=IDP     1439     A
DBG=IDP   ROL=X5   =      *+7         SAVED (X5)                                 000560  DBG=IDP     1440     A
DBG=IDP            LOC    *O                                                             DBG=IDP     1441     A
DBG=IDP   SNP      SPACE  4,8                                                            DBG=IDP     1442     A
DBG=IDP  **        SNP - SNAPSHOT OF CORE AND REGISTERS.                                 DBG=IDP     1443     A
DBG=IDP  *                                                                               DBG=IDP     1444     A
DBG=IDP  *                                                                               DBG=IDP     1445     A
DBG=IDP  *                CALLED BY *SNAP* MACRO.                                        DBG=IDP     1446     A
DBG=IDP  *                                                                               DBG=IDP     1447     A
DBG=IDP  *         LOWER HALF OF *RJ SNP=* WORD HAS FWA OF PARAMETER LIST--              DBG=IDP     1448     A
DBG=IDP  *                                                                               DBG=IDP     1449     A
DBG=IDP  * +       RJ     SNP=                                                           DBG=IDP     1450     A
DBG=IDP  * -       VFD    30/FWA OF PARAMETER LIST                                       DBG=IDP     1451     A
DBG=IDP  *                                                                               DBG=IDP     1452     A
DBG=IDP  *         PARAMETER LIST EXISTS IN LOCAL BLOCK *USE DEBUG*, AS SET UP BY        DBG=IDP     1453     A
DBG=IDP  *         BY *SNAP* MACRO. (SEE *DBG=MAC*) --                                   DBG=IDP     1454     A
DBG=IDP  *                                                                               DBG=IDP     1455     A
DBG=IDP  *         VFD    60/LOWER LIMIT (LL)                                            DBG=IDP     1456     A
DBG=IDP  *         VFD    60/UPPER LIMIT (UL)                                            DBG=IDP     1457     A
DBG=IDP  *         VFD    60/INCREMENT (INC)                                             DBG=IDP     1458     A
DBG=IDP  *         VFD    60/10HNAME  (OR .ZR. IF NO NAME)                               DBG=IDP     1459     A
DBG=IDP  *         VFD    60/0        (USED BY *FRK* TO KEEP SNAP COUNT)                 DBG=IDP     1460     A
DBG=IDP  *         VFD    18/0,21/LVL OF INDIRECT ADDRESSING,21/FWA                      DBG=IDP     1461     A
DBG=IDP  *         VFD    18/0,21/LVL OF INDIRECT ADDRESSING,21/LWA                      DBG=IDP     1462     A
DBG=IDP  *         VFD    18/0,21/LVL OF INDIRECT ADDRESSING,21/LEN                      DBG=IDP     1463     A
DBG=IDP  *         VFD    60/USER FLAGS                                              001180  DBG=IDP     1464     A
DBG=IDP  *         VFD    60/REGISTER DUMP FLAG                                          DBG=IDP     1465     A
DBG=IDP  *                 .                                                             DBG=IDP     1466     A
DBG=IDP  *                 .                                                             DBG=IDP     1467     A
DBG=IDP  *         VFD    60/REGISTER DUMP FLAG                                          DBG=IDP     1468     A
DBG=IDP  *                .ZR. = NO REGISTER DUMP  (OR END OF REGISTER LIST)             DBG=IDP     1469     A
DBG=IDP  *                .MI. = DUMP ALL REGISTERS                                      DBG=IDP     1470     A
DBG=IDP  *                .GT.0 = ADDR OF REGISTER SAVE WORD FOR REGISTER TO DMP         DBG=IDP     1471     A
DBG=IDP  *                                                                               DBG=IDP     1472     A
DBG=IDP  *         USES   NONE        (ALL REGISTERS ARE SAVED AND RESTORED)             DBG=IDP     1473     A
DBG=IDP  *                                                                               DBG=IDP     1474     A
DBG=IDP  *         CALLS  DAR,DCM,DSR,FRK,HDR,RSR=,SOB,SVR=,USO=(IF DEF)                 DBG=IDP     1475     A
DBG=IDP                                                                                  DBG=IDP     1476     A
DBG=IDP                                                                                  DBG=IDP     1477     A
DBG=IDP   SNP=     SUBR               ** ENTRY/EXIT **                                   DBG=IDP     1478     A
DBG=IDP            RJ     SVR=        SAVE ALL REGISTERS                                 DBG=IDP     1479     A
DBG=IDP            SX1    2           SET TO *EXECUTIVE IS /SNP=/*                       DBG=IDP     1480     A
DBG=IDP            RJ     SOB         SET OUTPUT BIT FLAGS                               DBG=IDP     1481     A
DBG=IDP            SA4    SNP=        (X4) = 30/EQ CALLING ADDRESS+1, 30/0               DBG=IDP     1482     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            LX4    30                                                          DBG=IDP    1483    A
DBG=IDP            SA5    X4-1        (A5,X5) = ADDR + CNTS OF CALLING *RJ*           DBG=IDP    1484    A
DBG=IDP            SX6    X5          (X6) = FWA OF PARAMETER LIST                    DBG=IDP    1485    A
DBG=IDP            SA6    FWAPARM                                                     DBG=IDP    1486    A
DBG=IDP                                                                     001220   DBG=IDP    1487    A
DBG=IDP   #USO     IF     DEF,USO=                                         001230   DBG=IDP    1488    A
DBG=IDP            RJ     =XUSO=      USER SNP= OWNCODE                    001240   DBG=IDP    1489    A
DBG=IDP            PL     X1,SNP4     IF NO SNAPSHOT THIS TIME             001250   DBG=IDP    1490    A
DBG=IDP   #USO     ENDIF                                                    001260   DBG=IDP    1491    A
DBG=IDP                                                                     001270   DBG=IDP    1492    A
DBG=IDP            SA5    FWAPARM     (X5) = FWA OF PARAMETER LIST         001280   DBG=IDP    1493    A
DBG=IDP            RJ     FRK         CHECK FREQUENCY PARAMETERS                      DBG=IDP    1494    A
DBG=IDP            ZR     X5,SNP4     IF NO SNAP THIS TIME                            DBG=IDP    1495    A
DBG=IDP            RJ     HDR         OUTPUT SNAP HEADER                              DBG=IDP    1496    A
DBG=IDP            SA1    X5+SN=SRL   (A1,X1) = ADDR + CNTS OF 1ST WORD OF            DBG=IDP    1497    A
DBG=IDP   *                              REGISTER LIST                               DBG=IDP    1498    A
DBG=IDP            PL     X1,SNP2     IF NOT DUMPING ALL REGISTERS                    DBG=IDP    1499    A
DBG=IDP            RJ     DAR         DUMP ALL REGISTERS                              DBG=IDP    1500    A
DBG=IDP            EQ     SNP3                                                        DBG=IDP    1501    A
DBG=IDP                                                                               DBG=IDP    1502    A
DBG=IDP   SNP2     RJ     DSR         DUMP SELECTED REGISTERS                         DBG=IDP    1503    A
DBG=IDP                                                                               DBG=IDP    1504    A
DBG=IDP   SNP3     SA5    FWAPARM                                                     DBG=IDP    1505    A
DBG=IDP            SA2    X5+SN=FWA   (X2) = 10/0,21/LVL OF IND ADDR,21/FWA           DBG=IDP    1506    A
DBG=IDP            SA3    A2+B1       (X3) = 18/0,21/LVL OF IND ADDR,21/LWA           DBG=IDP    1507    A
DBG=IDP            SA4    A3+B1       (X4) = 18/0,21/LVL OF IND ADDR,21/LEN           DBG=IDP    1508    A
DBG=IDP            RJ     DCM         DUMP CENTRAL MEMORY                             DBG=IDP    1509    A
DBG=IDP                                                                               DBG=IDP    1510    A
DBG=IDP   SNP4     RJ     =XRSR=      RESTORE ALL REGISTERS                           DBG=IDP    1511    A
DBG=IDP            EQ     EXIT.                                                       DBG=IDP    1512    A
DBG=IDP   SOB      SPACE  4,8                                                         DBG=IDP    1513    A
DBG=IDP   **       SOB - SET OUTPUT BIT FLAGS.                                        DBG=IDP    1514    A
DBG=IDP   *                                                                           DBG=IDP    1515    A
DBG=IDP   *                                                                           DBG=IDP    1516    A
DBG=IDP   *                THIS ROUTINE SETS BITS IN THE MASTER CONTROL FLAG,         DBG=IDP    1517    A
DBG=IDP   *        *IDPFLG*, WHICH DETERMINES WHERE OUTPUT IS TO BE WRITTEN BY A      DBG=IDP    1518    A
DBG=IDP   *        BATCH (I.E. NON-INTERACTIVE) EXECUTIVE --                          DBG=IDP    1519    A
DBG=IDP   *                                                                           DBG=IDP    1520    A
DBG=IDP   *          1. IF INTERACTIVE MODE HAS EVER BEEN ENTERED (BIT *FTO* ON),     DBG=IDP    1521    A
DBG=IDP   *                THEN NO FLAGS ARE SET  (I.E. ALL FLAGS ARE AS SET BY       DBG=IDP    1522    A
DBG=IDP   *                INTERACTIVE USER VIA *OUTPUT* COMMAND).                    DBG=IDP    1523    A
DBG=IDP   *          2. IF INTERACTIVE MODE HAS NOT YET BEEN ENTERED                  DBG=IDP    1524    A
DBG=IDP   *                (BIT *FTO* OFF), THEN SET --                               DBG=IDP    1525    A
DBG=IDP   *                  BDO = WRITING TO BATCH DEBUG OUTPUT FILE, *F.BDO*        DBG=IDP    1526    A
DBG=IDP   *                  SNL = L.BSL                                              DBG=IDP    1527    A
DBG=IDP   *                                                                           DBG=IDP    1528    A
DBG=IDP   *        ENTRY  (X1) = BITS 0 THRU *IDF.XECL-1* CONTAIN THE VALUE TO        DBG=IDP    1529    A
DBG=IDP   *                      BE PLACED IN *XEC* FIELD,                            DBG=IDP    1530    A
DBG=IDP   *                      (BITS 59 THRU *IDF.XECL* ARE IGNORED).               DBG=IDP    1531    A
DBG=IDP   *                                                                           DBG=IDP    1532    A
DBG=IDP   *        EXIT   (IDPFLG) UPDATED                                            DBG=IDP    1533    A
DBG=IDP   *                                                                           DBG=IDP    1534    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *         USES   X - 1,2,3,4,6                                  DBG=IDP   1535   A
DBG=IDP  *                A - 2,6                                        DBG=IDP   1536   A
DBG=IDP  *                B - NONE                                       DBG=IDP   1537   A
DBG=IDP  *                                                               DBG=IDP   1538   A
DBG=IDP  *         CALLS  NONE                                           DBG=IDP   1539   A
DBG=IDP                                                                  DBG=IDP   1540   A
DBG=IDP                                                                  DBG=IDP   1541   A
DBG=IDP   SOB      SUBR               ** ENTRY/EXIT **                   DBG=IDP   1542   A
DBG=IDP            SA2    IDPFLG                                         DBG=IDP   1543   A
DBG=IDP            MX3    -IDF.XECL                                      DBG=IDP   1544   A
DBG=IDP            SX4    L.BSL                                          DBG=IDP   1545   A
DBG=IDP            BX1    -X3*X1                                         DBG=IDP   1546   A
DBG=IDP            LX2    0-IDF.XECP                                     DBG=IDP   1547   A
DBG=IDP            BX2    X3*X2       CLEAR OLD *XEC*                    DBG=IDP   1548   A
DBG=IDP            IX6    X2+X1       MERGE NEW *XEC*                    DBG=IDP   1549   A
DBG=IDP            LX2    59-IDF.FTOP+IDF.XECP-0                         DBG=IDP   1550   A
DBG=IDP            LX6    IDF.XECP-0                                     DBG=IDP   1551   A
DBG=IDP            MX3    1                                              DBG=IDP   1552   A
DBG=IDP            SA6    A2                                             DBG=IDP   1553   A
DBG=IDP            MI     X2,EXIT.    IF *IDP* HAS EVER BEEN CALLED      DBG=IDP   1554   A
DBG=IDP            LX3    IDF.BDOP-59                                    DBG=IDP   1555   A
DBG=IDP            MX2    -IDF.SNLL                                      DBG=IDP   1556   A
DBG=IDP            BX6    X6+X3       MERGE *BDO*                        DBG=IDP   1557   A
DBG=IDP            LX2    IDF.SNLP-0                                     DBG=IDP   1558   A
DBG=IDP            LX4    IDF.SNLP-0                                     DBG=IDP   1559   A
DBG=IDP            BX6    X2*X6       CLEAR OLD *SNL*                    DBG=IDP   1560   A
DBG=IDP            IX6    X6+X4       MERGE NEW *SNL*                    DBG=IDP   1561   A
DBG=IDP            SA6    A2                                             DBG=IDP   1562   A
DBG=IDP            EQ     EXIT.                                          DBG=IDP   1563   A
DBG=IDP   WOD      SPACE  4,8                                            DBG=IDP   1564   A
DBG=IDP  **        WOD - CONVERT A WORD TO OCTAL DISPLAY CODE.           DBG=IDP   1565   A
DBG=IDP  *                                                               DBG=IDP   1566   A
DBG=IDP  *                                                               DBG=IDP   1567   A
DBG=IDP  *         ENTRY  (X1)   =  BINARY WORD TO CONVERT               DBG=IDP   1568   A
DBG=IDP  *                                                               DBG=IDP   1569   A
DBG=IDP  *         EXIT   (X6)   =  UPPER 10 OCTAL DIGITS OF CONVERTED WORD  DBG=IDP   1570   A
DBG=IDP  *                (X7)   =  LOWER 10 OCTAL DIGITS OF CONVERTED WORD  DBG=IDP   1571   A
DBG=IDP  *                                                               DBG=IDP   1572   A
DBG=IDP  *         USES   X - ALL                                        DBG=IDP   1573   A
DBG=IDP  *                A - 2,3,4,5                                    DBG=IDP   1574   A
DBG=IDP  *                B - NONE                                       DBG=IDP   1575   A
DBG=IDP  *                                                               DBG=IDP   1576   A
DBG=IDP  *         CALLS  NONE                                           DBG=IDP   1577   A
DBG=IDP                                                                  DBG=IDP   1578   A
DBG=IDP                                                                  DBG=IDP   1579   A
DBG=IDP   #WOD     IF     -DEF,WOD                                       DBG=IDP   1580   A
DBG=IDP                                                                  DBG=IDP   1581   A
DBG=IDP   WOD      SUBR               ** ENTRY/EXIT **                   DBG=IDP   1582   A
DBG=IDP                                                                  DBG=IDP   1583   A
DBG=IDP  *         ON ENTRY   (X1)  = ABCDE FGHIJ KLMNO PQRST            DBG=IDP   1584   A
DBG=IDP                                                                  DBG=IDP   1585   A
DBG=IDP            SA2    WODA        7.... 7.... 7.... 7....            DBG=IDP   1586   A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP          BX7    X2*X1       A.... F.... K.... P....                          DBG=IDP    1587    A
DBG=IDP          LX1    3           BCDEF GHIJK LMNOP QRSTA                          DBG=IDP    1588    A
DBG=IDP          BX6    X2*X1       B.... G.... L.... Q....                          DBG=IDP    1589    A
DBG=IDP          LX1    3           CDEFG HIJKL MNOPQ RSTAB                          DBG=IDP    1590    A
DBG=IDP          LX7    9*3         .K... .P... .A... .F...                          DBG=IDP    1591    A
DBG=IDP          BX0    X2*X1       C.... H.... M.... R....                          DBG=IDP    1592    A
DBG=IDP          LX6    7*3         ...L. ...Q. ...B. ...G.                          DBG=IDP    1593    A
DBG=IDP          LX1    3           DEFGH IJKLM NOPQR STABC                          DBG=IDP    1594    A
DBG=IDP          IX7    X6+X7       .K.L. .P.Q. .A.B. .F.G.                          DBG=IDP    1595    A
DBG=IDP          BX5    X2*X1       D.... I.... N.... S....                          DBG=IDP    1596    A
DBG=IDP          LX1    3           EFGHI JKLMN OPQRS TABCD                          DBG=IDP    1597    A
DBG=IDP          LX0    5*3         H.... M.... R.... C....                          DBG=IDP    1598    A
DBG=IDP          BX6    X2*X1       E.... J.... O.... T....                          DBG=IDP    1599    A
DBG=IDP          IX7    X7+X0       HK.L. MP.Q. RA.B. CF.G.                          DBG=IDP    1600    A
DBG=IDP          LX5    3*3         ..I.. ..N.. ..S.. ..D..                          DBG=IDP    1601    A
DBG=IDP          LX6    3           ....J ....O ....T ....E                          DBG=IDP    1602    A
DBG=IDP          IX7    X7+X6       HK.LJ MP.QO RA.BT CF.GE                          DBG=IDP    1603    A
DBG=IDP          SA4    A2+B1       .7.7. 7.7.7 ..... .....                          DBG=IDP    1604    A
DBG=IDP          SA3    A4+B1       ..... .7.7. 7.7.7 .....                          DBG=IDP    1605    A
DBG=IDP          BX7    X7+X5       HKILJ MPNQO RASBT CFDGE                          DBG=IDP    1606    A
DBG=IDP          BX2    X7*X4       .K.L. M.N.O ..... .....                          DBG=IDP    1607    A
DBG=IDP          SA5    A3+B1       00000 00000 00000 00000                          DBG=IDP    1608    A
DBG=IDP          BX1    X7*X3       ..... .P.Q. R.S.T .....                          DBG=IDP    1609    A
DBG=IDP          LX7    10*3        RASBT CFDGE HKILJ MPNQO                          DBG=IDP    1610    A
DBG=IDP          IX2    X5+X2       .K.L. M.N.O 00000 00000                          DBG=IDP    1611    A
DBG=IDP          BX0    X7*X4       .A.B. C.D.E ..... .....                          DBG=IDP    1612    A
DBG=IDP          IX0    X0+X5       .A.B. C.D.E 00000 00000                          DBG=IDP    1613    A
DBG=IDP          LX1    15*3        ..... ..... .P.Q. R.S.T                          DBG=IDP    1614    A
DBG=IDP          BX3    X7*X3       ..... .F.G. H.I.J .....                          DBG=IDP    1615    A
DBG=IDP          IX7    X1+X2       .K.L. M.N.O .P.Q. R.S.T                          DBG=IDP    1616    A
DBG=IDP          LX3    15*3        ..... ..... .F.G. H.I.J                          DBG=IDP    1617    A
DBG=IDP          IX6    X0+X3       .A.B. C.D.E .F.G. H.I.J                          DBG=IDP    1618    A
DBG=IDP          EQ     EXIT.                                                        DBG=IDP    1619    A
DBG=IDP                                                                              DBG=IDP    1620    A
DBG=IDP   WODA   CON    7000070000700007000 0B                                       DBG=IDP    1621    A
DBG=IDP          CON    0707070707000000000 0B                                       DBG=IDP    1622    A
DBG=IDP          CON    0000007070707070000 0B                                       DBG=IDP    1623    A
DBG=IDP          CON    10H0000000000                                                DBG=IDP    1624    A
DBG=IDP   #WOD   ENDIF                                                               DBG=IDP    1625    A
DBG=IDP   IDP    TITLE  IDP - INTERACTIVE DEBUGGING PACKAGE.                         DBG=IDP    1626    A
DBG=IDP  **      IDP - INTERACTIVE DEBUGGING PACKAGE.                                DBG=IDP    1627    A
DBG=IDP  *                                                                           DBG=IDP    1628    A
DBG=IDP  *                                                                           DBG=IDP    1629    A
DBG=IDP  *               THIS IS THE INTERACTIVE DEBUGGING PACKAGE WHICH ALLOWS      DBG=IDP    1630    A
DBG=IDP  *       ITS USER THE JOYS AND SORROWS OF INTERACTIVE DEBUGGING.             DBG=IDP    1631    A
DBG=IDP  *                                                                           DBG=IDP    1632    A
DBG=IDP  *               INITIAL ENTRY IS VIA *BREAK* MACRO (DBG=MAC), WHICH         DBG=IDP    1633    A
DBG=IDP  *       ASSEMBLES AN *RJ IDP=* AT THE DESIRED LOCATION TO BE BREAK-         DBG=IDP    1634    A
DBG=IDP  *       POINTED. SUBSEQUENT ENTRIES CAN BE VIA MORE ASSEMBLED *BREAK*S      DBG=IDP    1635    A
DBG=IDP  *       OR BY PLACING AN *IDP* GENERATED BREAKPOINT, VIA *BREAK*            DBG=IDP    1636    A
DBG=IDP  *       COMMAND, AT THE DESIRED LOCATION TO BE BREAKPOINTED.                DBG=IDP    1637    A
DBG=IDP  *                                                                           DBG=IDP    1638    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *        IDP COMMANDS ARE--                                          DBG=IDP    1639    A
DBG=IDP  *                                                                    DBG=IDP    1640    A
DBG=IDP  *        BREAK                                                       DBG=IDP    1641    A
DBG=IDP  *        BREAK  ADDR,LL,UL,INC                                       DBG=IDP    1642    A
DBG=IDP  *        BRPL   ADDR,A1,LL,UL,INC                                    DBG=IDP    1643    A
DBG=IDP  *        BRMI   ADDR,A1,LL,UL,INC                                    DBG=IDP    1644    A
DBG=IDP  *        BRZR   ADDR,A1,LL,UL,INC                                    DBG=IDP    1645    A
DBG=IDP  *        BRNZ   ADDR,A1,LL,UL,INC                                    DBG=IDP    1646    A
DBG=IDP  *        BREQ   ADDR,A1,A2,LL,UL,INC                                 DBG=IDP    1647    A
DBG=IDP  *        BRNE   ADDR,A1,A2,LL,UL,INC                                 DBG=IDP    1648    A
DBG=IDP  *        BRLT   ADDR,A1,A2,LL,UL,INC                                 DBG=IDP    1649    A
DBG=IDP  *        BRGE   ADDR,A1,A2,LL,UL,INC                                 DBG=IDP    1650    A
DBG=IDP  *        BRLE   ADDR,A1,A2,LL,UL,INC                                 DBG=IDP    1651    A
DBG=IDP  *        BRGT   ADDR,A1,A2,LL,UL,INC                                 DBG=IDP    1652    A
DBG=IDP  *        CODE   FWA,LWA,LEN                                          DBG=IDP    1653    A
DBG=IDP  *        CONNECT FILE                                                DBG=IDP    1654    A
DBG=IDP  *        DISCONT FILE                                                DBG=IDP    1655    A
DBG=IDP  *        DPC FWA,LWA,LEN                                             DBG=IDP    1656    A
DBG=IDP  *        END                                                         DBG=IDP    1657    A
DBG=IDP  *        FREEZE                                            000220    DBG=IDP    1658    A
DBG=IDP  *        JUMP   ADDR                                                 DBG=IDP    1659    A
DBG=IDP  *        OUTPUT I,B,IE,BE                                            DBG=IDP    1660    A
DBG=IDP  *        REGS                                                        DBG=IDP    1661    A
DBG=IDP  *        REGS   R1,R2,...,RN                                         DBG=IDP    1662    A
DBG=IDP  *        SET                                                         DBG=IDP    1663    A
DBG=IDP  *        SET    NAME,VALUE                                           DBG=IDP    1664    A
DBG=IDP  *        SNAP   FWA,LWA,LEN                                          DBG=IDP    1665    A
DBG=IDP  *        STORE  ADDR,C1,C2,C3,C4                                     DBG=IDP    1666    A
DBG=IDP  *        STEP   L,RJ                                                 DBG=IDP    1667    A
DBG=IDP  *        STPL   ADDR,L,RJ                                            DBG=IDP    1668    A
DBG=IDP  *        STMI   ADDR,L,RJ                                            DBG=IDP    1669    A
DBG=IDP  *        STZR   ADDR,L,RJ                                            DBG=IDP    1670    A
DBG=IDP  *        STNZ   ADDR,L,RJ                                            DBG=IDP    1671    A
DBG=IDP  *        STEQ   ADDR1,ADDR2,L,RJ                                     DBG=IDP    1672    A
DBG=IDP  *        STNE   ADDR1,ADDR2,L,RJ                                     DBG=IDP    1673    A
DBG=IDP  *        STLT   ADDR1,ADDR2,L,RJ                                     DBG=IDP    1674    A
DBG=IDP  *        STGE   ADDR1,ADDR2,L,RJ                                     DBG=IDP    1675    A
DBG=IDP  *        STLE   ADDR1,ADDR2,L,RJ                                     DBG=IDP    1676    A
DBG=IDP  *        STGT   ADDR1,ADDR2,L,RJ                                     DBG=IDP    1677    A
DBG=IDP  *        STRANGE ADDR1,ADDR2,L,RJ                                    DBG=IDP    1678    A
DBG=IDP  *        STNR   NR,L,RJ                                              DBG=IDP    1679    A
DBG=IDP  *        STAR   ADDR1,ADDR2,L,RJ                                     DBG=IDP    1680    A
DBG=IDP  *        UNBREAK ADDR1,...,ADDRN                                     DBG=IDP    1681    A
DBG=IDP  *        UNSET  NAME1,...,NAMEN                                      DBG=IDP    1682    A
DBG=IDP  *        WHERE  ADDR1,...,ADDRN                                      DBG=IDP    1683    A
DBG=IDP  *        XEQ                                                         DBG=IDP    1684    A
DBG=IDP  *                                                                    DBG=IDP    1685    A
DBG=IDP  *        MAY THE GODS BE WITH YOU...                                 DBG=IDP    1686    A
DBG=IDP  *                                                                    DBG=IDP    1687    A
DBG=IDP  *        ENTRY  NONE                                                 DBG=IDP    1688    A
DBG=IDP  *                                                                    DBG=IDP    1689    A
DBG=IDP  *        EXIT   NONE                                                 DBG=IDP    1690    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *                                                                    DBG=IDP    1691    A
DBG=IDP  *         USES   NONE        (OR IT WOULDNT BE ANY GOOD)            DBG=IDP    1692    A
DBG=IDP  *                                                                    DBG=IDP    1693    A
DBG=IDP  *         CALLS  (EXTERNAL TO DBG=IDP)--                            DBG=IDP    1694    A
DBG=IDP  *                CDD,COD,DXB,OPEN,READC,RSR=,SFN,SVR=,WRITEC,WRITER     000180 DBG=IDP    1695    A
DBG=IDP                                                                       DBG=IDP    1696    A
DBG=IDP                                                                       DBG=IDP    1697    A
DBG=IDP   #OS      IFNE   .OS,2       IF NOT SCOPE 2                         DBG=IDP    1698    A
DBG=IDP                                                                       DBG=IDP    1699    A
DBG=IDP   IDP=     SUBR               ** ENTRY **                            DBG=IDP    1700    A
DBG=IDP            RJ     =XSVR=      SAVE ALL REGISTTERS                    DBG=IDP    1701    A
DBG=IDP                                                                       DBG=IDP    1702    A
DBG=IDP  *         INITIALIZATION.                                           DBG=IDP    1703    A
DBG=IDP                                                                       DBG=IDP    1704    A
DBG=IDP            SA1    IDP=                                               DBG=IDP    1705    A
DBG=IDP            SX6    60                                                 DBG=IDP    1706    A
DBG=IDP            LX1    30                                                 DBG=IDP    1707    A
DBG=IDP            SX7    X1                                                 DBG=IDP    1708    A
DBG=IDP            SA6    IDPPOS      SET UP POS COUNTER                     DBG=IDP    1709    A
DBG=IDP            SA7    IDPPREG     SET UP PSEUDO P REGISTER               DBG=IDP    1710    A
DBG=IDP            SB6    IDPXFT                                             DBG=IDP    1711    A
DBG=IDP            RJ     CLZ         CLEAR *IDP* TABLE -- TRANSFER ADDRESSES DBG=IDP   1712    A
DBG=IDP                                                                       DBG=IDP    1713    A
DBG=IDP  *         SET UP *IDPFLG* - MASTER CONTROL FLAG.                    DBG=IDP    1714    A
DBG=IDP  *                                                                    DBG=IDP    1715    A
DBG=IDP  *           SET  XEC = *IDP=*                                       DBG=IDP    1716    A
DBG=IDP  *           IF 1ST TIME *IDP* HAS BEEN ENTERED (BIT *FTO* OFF), SET -- DBG=IDP 1717    A
DBG=IDP  *                FTO = 1ST TIME ONLY CODE HAS BEEN EXECUTED         DBG=IDP    1718    A
DBG=IDP  *                IDO = WRITING INTERACTIVELY (F.IDO)                DBG=IDP    1719    A
DBG=IDP  *                SNL = L.ISL                                        DBG=IDP    1720    A
DBG=IDP                                                                       DBG=IDP    1721    A
DBG=IDP            SA1    IDPFLG                                             DBG=IDP    1722    A
DBG=IDP            MX2    -IDF.XECL                                          DBG=IDP    1723    A
DBG=IDP            MX3    1                                                  DBG=IDP    1724    A
DBG=IDP            LX2    IDF.XECP-0                                         DBG=IDP    1725    A
DBG=IDP            BX6    X2*X1       CLEAR OLD *XEC*/SET NEW *XEC*          DBG=IDP    1726    A
DBG=IDP            MX4    -IDF.SNLL                                          DBG=IDP    1727    A
DBG=IDP            LX1    59-IDF.FTOP                                        DBG=IDP    1728    A
DBG=IDP            SA6    A1                                                 DBG=IDP    1729    A
DBG=IDP            MI     X1,IDP2     IF NOT 1ST TIME                        DBG=IDP    1730    A
DBG=IDP            LX4    IDF.SNLP-0                                         DBG=IDP    1731    A
DBG=IDP            LX3    IDF.FTOP-59                                        DBG=IDP    1732    A
DBG=IDP            BX6    X4*X6       CLEAR OLD *SNL*                        DBG=IDP    1733    A
DBG=IDP            =X1    L.ISL                                              DBG=IDP    1734    A
DBG=IDP            BX6    X6+X3       MERGE *FTO*                            DBG=IDP    1735    A
DBG=IDP            LX3    IDF.IDOP-IDF.FTOP                                  DBG=IDP    1736    A
DBG=IDP            BX6    X6+X1       MERGE NEW *SNL*                        DBG=IDP    1737    A
DBG=IDP            IX6    X6+X3       MERGE *IDO*                            DBG=IDP    1738    A
DBG=IDP            SA6    A1                                                 DBG=IDP    1739    A
DBG=IDP                                                                       DBG=IDP    1740    A
DBG=IDP  *         SET UP INTERACTIVE FILES.                                 000240 DBG=IDP    1741    A
DBG=IDP                                                                      000250 DBG=IDP    1742    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            RJ     IIF         INITIALIZE INTERACTIVE FILES              000260   DBG=IDP     1743     A
DBG=IDP                                                                                   DBG=IDP     1744     A
DBG=IDP  *         CLEAR/INITIALIZE TABLES.                                               DBG=IDP     1745     A
DBG=IDP                                                                                   DBG=IDP     1746     A
DBG=IDP            SB6    IDPBA                                                           DBG=IDP     1747     A
DBG=IDP            RJ     CLZ         CLEAR *IDP* TABLE -- BREAK ADDRESSES               DBG=IDP     1748     A
DBG=IDP            SB6    IDPSET                                                          DBG=IDP     1749     A
DBG=IDP            RJ     CLZ         CLEAR *IDP* TABLE -- SET NAMES                     DBG=IDP     1750     A
DBG=IDP            RJ     IST         (RE)INITIALIZE *SET* TABLE                         DBG=IDP     1751     A
DBG=IDP                                                                                   DBG=IDP     1752     A
DBG=IDP  *         INITIAL BREAK PROCESSING.                                              DBG=IDP     1753     A
DBG=IDP                                                                                   DBG=IDP     1754     A
DBG=IDP   IDP2     SA1    IDPPREG                                                         DBG=IDP     1755     A
DBG=IDP            SA5    X1-1        (X5) = 30/RJ IDP, 30/FWA PARAMETER LIST            DBG=IDP     1756     A
DBG=IDP            BX6    X6-X6                                                           DBG=IDP     1757     A
DBG=IDP            SX5    X5                                                              DBG=IDP     1758     A
DBG=IDP            SA6    ST=ENDX     SET TO *DEFAULT IS NOT IDP GENERATED BRK*          DBG=IDP     1759     A
DBG=IDP            RJ     BRK         BREAK PROCESSOR                                    DBG=IDP     1760     A
DBG=IDP                                                                                   DBG=IDP     1761     A
DBG=IDP   #UIO     IF     DEF,UIO=                                                        DBG=IDP     1762     A
DBG=IDP            RJ     =XUIO=      USER IDP OWNCODE                                   DBG=IDP     1763     A
DBG=IDP            PL     X1,ST=ENDW  IF NO INTERACTIVE BREAK THIS TIME                  DBG=IDP     1764     A
DBG=IDP   #UIO     ENDIF                                                                  DBG=IDP     1765     A
DBG=IDP                                                                                   DBG=IDP     1766     A
DBG=IDP            ZR     X5,ST=ENDW  IF CONDITIONAL BREAK NOT SATISFIED                 DBG=IDP     1767     A
DBG=IDP            RJ     FRK         CHECK FREQUENCY PARAMETERS                         DBG=IDP     1768     A
DBG=IDP            ZR     X5,ST=ENDW  IF NO BREAK THIS TIME                              DBG=IDP     1769     A
DBG=IDP                                                                                   DBG=IDP     1770     A
DBG=IDP  *         FLUSH *F.BDO - BATCH DEBUG OUTPUT FILE*.                               DBG=IDP     1771     A
DBG=IDP                                                                                   DBG=IDP     1772     A
DBG=IDP            WRITER =XF.BDO,,RCL                                                    DBG=IDP     1773     A
DBG=IDP                                                                                   DBG=IDP     1774     A
DBG=IDP  *         INTRODUCE YOURSELF.                                                    DBG=IDP     1775     A
DBG=IDP                                                                                   DBG=IDP     1776     A
DBG=IDP   IDP2A    BSS    0                                                               DBG=IDP     1777     A
DBG=IDP            SA1    =10H **IDP**                                                    DBG=IDP     1778     A
DBG=IDP            SA2    =10H CALLD BY                                                   DBG=IDP     1779     A
DBG=IDP            SA3    X5+SN=HDR                                                       DBG=IDP     1780     A
DBG=IDP            BX6    X1                                                              DBG=IDP     1781     A
DBG=IDP            LX7    X2                                                              DBG=IDP     1782     A
DBG=IDP            SA6    SNAPLNE                                                         DBG=IDP     1783     A
DBG=IDP            SA7    A6+B1                                                           DBG=IDP     1784     A
DBG=IDP            BX6    X3                                                              DBG=IDP     1785     A
DBG=IDP            ZR     X3,IDP4     IF NO HEADER PROVIDED BY CALLER .OR.               DBG=IDP     1786     A
DBG=IDP  *                               IDP GENERATED BREAK                              DBG=IDP     1787     A
DBG=IDP            SA6    A6+B1                                                           DBG=IDP     1788     A
DBG=IDP            SA7    A6+B1                                                           DBG=IDP     1789     A
DBG=IDP                                                                                   DBG=IDP     1790     A
DBG=IDP   IDP4     SA1    IDPPREG                                                         DBG=IDP     1791     A
DBG=IDP            SX1    X1-1                                                            DBG=IDP     1792     A
DBG=IDP                                                                                   DBG=IDP     1793     A
DBG=IDP   #FRA     IF     DEF,FRA=                                                        DBG=IDP     1794     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            SB7    IDP5        (B7) = RETURN ADDR FOR *FRA=*                DBG=IDP    1795    A
DBG=IDP            EQ     =XFRA=      FIND RELATIVE ADDRESS                        DBG=IDP    1796    A
DBG=IDP                                                                            DBG=IDP    1797    A
DBG=IDP   IDP5     BSS    0                                                        DBG=IDP    1798    A
DBG=IDP                                                                            DBG=IDP    1799    A
DBG=IDP   #FRA     ELSE                                                            DBG=IDP    1800    A
DBG=IDP            RJ     =XCOD       CONVERT BINARY TO OCTAL DPC                  DBG=IDP    1801    A
DBG=IDP            BX6    X4                                                       DBG=IDP    1802    A
DBG=IDP            MX7    0                                                        DBG=IDP    1803    A
DBG=IDP   #FRA     ENDIF                                                           DBG=IDP    1804    A
DBG=IDP                                                                            DBG=IDP    1805    A
DBG=IDP            SA6    A7+B1                                                    DBG=IDP    1806    A
DBG=IDP            SA7    A6+B1                                                    DBG=IDP    1807    A
DBG=IDP            PRIDP  SNAPLNE                                                  DBG=IDP    1808    A
DBG=IDP            EQ     IDP8        READ 1ST LINE...                             DBG=IDP    1809    A
DBG=IDP   IDP=MN   SPACE  4,8                                                      DBG=IDP    1810    A
DBG=IDP  *         ** MAIN LOOP NODE **                                            DBG=IDP    1811    A
DBG=IDP                                                                            DBG=IDP    1812    A
DBG=IDP   IDP=MN   BSS    0                                                        DBG=IDP    1813    A
DBG=IDP            SA1    IDPSTMT                                                  DBG=IDP    1814    A
DBG=IDP            PL     X1,IDP7     IF LAST STMT PROCESSED WAS NOT IN *PROC*     DBG=IDP    1815    A
DBG=IDP                                                                            DBG=IDP    1816    A
DBG=IDP  *         HERE IF LAST STATEMENT WAS IN A PROCEDURE.                      DBG=IDP    1817    A
DBG=IDP                                                                            DBG=IDP    1818    A
DBG=IDP            EQ     *+4S15      OOPS...                                      DBG=IDP    1819    A
DBG=IDP                                                                            DBG=IDP    1820    A
DBG=IDP  *         CHECK FOR AVAILABILITY OF NEXT STATEMENT.                       DBG=IDP    1821    A
DBG=IDP                                                                            DBG=IDP    1822    A
DBG=IDP   IDP7     SA2    X1          (X2) = LEN WORD FOR LAST STMT PROCESSED      DBG=IDP    1823    A
DBG=IDP            LX2    59-TB.LASP                                               DBG=IDP    1824    A
DBG=IDP            MI     X2,IDP8     IF LAST STMT PROCESSED WAS LAST ON LINE      DBG=IDP    1825    A
DBG=IDP            LX2    0-TB.LENP+TB.LASP-59                                     DBG=IDP    1826    A
DBG=IDP            SX1    X1                                                       DBG=IDP    1827    A
DBG=IDP            SX2    X2+B1                                                    DBG=IDP    1828    A
DBG=IDP            IX6    X1+X2       (X6) = ADDR OF LEN WORD FOR NEXT STMT        DBG=IDP    1829    A
DBG=IDP            SA6    IDPSTMT                                                  DBG=IDP    1830    A
DBG=IDP            SA5    X6+B1       (A5,X5) = A+C OF 1ST TOKEN IN NEXT STMT      DBG=IDP    1831    A
DBG=IDP            EQ     IDP12                                                    DBG=IDP    1832    A
DBG=IDP                                                                            DBG=IDP    1833    A
DBG=IDP  *         HERE TO READ NEW LINE.                                          DBG=IDP    1834    A
DBG=IDP                                                                            DBG=IDP    1835    A
DBG=IDP   IDP8     SA1    IDPFLG                                                   DBG=IDP    1836    A
DBG=IDP            LX1    59-IDF.INPP                                              DBG=IDP    1837    A
DBG=IDP            MI     X1,IDP9     IF READING FROM BATCH INPUT (F.BDI)          DBG=IDP    1838    A
DBG=IDP                                                                            DBG=IDP    1839    A
DBG=IDP            DUP    L.PRB,1                                                  DBG=IDP    1840    A
DBG=IDP            WRITEC =XF.IDO,(=C=  =)                                         DBG=IDP    1841    A
DBG=IDP                                                                            DBG=IDP    1842    A
DBG=IDP            WRITEC =XF.IDO,(=C= "PROMPT"=)                                  DBG=IDP    1843    A
DBG=IDP            WRITER =XF.IDO,,RCL                                             DBG=IDP    1844    A
DBG=IDP                                                                            DBG=IDP    1845    A
DBG=IDP   IDP9     RJ     RIL         READ IDP LINE                                DBG=IDP    1846    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            SA2    IDPFLG                                                  DBG=IDP    1847    A
DBG=IDP            ZR     X1,IDP10    IF EOR/EOF NOT ENCOUNTERED                  DBG=IDP    1848    A
DBG=IDP            MX6    1                                                       DBG=IDP    1849    A
DBG=IDP            LX6    IDF.INPP-59                                             DBG=IDP    1850    A
DBG=IDP            BX6    -X6*X2      CLEAR INP                                   DBG=IDP    1851    A
DBG=IDP            SA6    A2                                                      DBG=IDP    1852    A
DBG=IDP            EQ     IDP8                                                    DBG=IDP    1853    A
DBG=IDP                                                                           DBG=IDP    1854    A
DBG=IDP  *         HERE TO ECHO LINE TO BATCH DEBUG OUTPUT FILE (F.BDO),          DBG=IDP    1855    A
DBG=IDP  *           IF REQUESTED.                                                DBG=IDP    1856    A
DBG=IDP                                                                           DBG=IDP    1857    A
DBG=IDP   IDP10    BX3    X2                                                      DBG=IDP    1858    A
DBG=IDP            LX2    59-IDF.BKOP                                             DBG=IDP    1859    A
DBG=IDP            LX3    59-IDF.BDOP                                             DBG=IDP    1860    A
DBG=IDP            BX4    X2*X3                                                   DBG=IDP    1861    A
DBG=IDP            PL     X4,IDP10A   IF NO ECHO REQUESTED .OR. NOT WRITING TO    DBG=IDP    1862    A
DBG=IDP  *                               BATCH DEBUG OUTPUT FILE (F.BDO)          DBG=IDP    1863    A
DBG=IDP            PRBDO  (=C=  =),1                                              DBG=IDP    1864    A
DBG=IDP            SA2    L=LIM                                                   DBG=IDP    1865    A
DBG=IDP            PRBDO  IDPLIM-1,X2                                             DBG=IDP    1866    A
DBG=IDP                                                                           DBG=IDP    1867    A
DBG=IDP  *         HERE TO ECHO LINE TO INTERACTIVE DEBUG OUTPUT FILE (F.IDO),    DBG=IDP    1868    A
DBG=IDP  *           IF REQUESTED.                                                DBG=IDP    1869    A
DBG=IDP                                                                           DBG=IDP    1870    A
DBG=IDP   IDP10A   SA1    IDPFLG                                                  DBG=IDP    1871    A
DBG=IDP            BX2    X1                                                      DBG=IDP    1872    A
DBG=IDP            LX1    59-IDF.IKOP                                             DBG=IDP    1873    A
DBG=IDP            LX2    59-IDF.IDOP                                             DBG=IDP    1874    A
DBG=IDP            BX3    X1*X2                                                   DBG=IDP    1875    A
DBG=IDP            PL     X3,IDP11    IF NO ECHO REQUESTED .OR. NOT WRITING TO    DBG=IDP    1876    A
DBG=IDP  *                               INTERACTIVE DEBUG OUTPUT FILE (F.IDO)    DBG=IDP    1877    A
DBG=IDP                                                                           DBG=IDP    1878    A
DBG=IDP            DUP    L.PRB,1                                                 DBG=IDP    1879    A
DBG=IDP            WRITEC =XF.IDO,(=C=  =)                                        DBG=IDP    1880    A
DBG=IDP                                                                           DBG=IDP    1881    A
DBG=IDP            WRITEC =XF.IDO,IDPLIM-1                                        DBG=IDP    1882    A
DBG=IDP                                                                           DBG=IDP    1883    A
DBG=IDP  *         HERE TO ENTOKEN NEW LINE.                                      DBG=IDP    1884    A
DBG=IDP                                                                           DBG=IDP    1885    A
DBG=IDP   IDP11    SX7    L.TB                                                    DBG=IDP    1886    A
DBG=IDP            SA6    IDPTB       (A6) = ADDR OF LEN WORD FOR 1ST STMT        DBG=IDP    1887    A
DBG=IDP            SA7    BUB=LEN                                                 DBG=IDP    1888    A
DBG=IDP            RJ     BUB         BURST/BUILD NEW LINE                        DBG=IDP    1889    A
DBG=IDP            SX6    IDPTB                                                   DBG=IDP    1890    A
DBG=IDP            SA5    IDPTB+1                                                 DBG=IDP    1891    A
DBG=IDP            SA6    IDPSTMT                                                 DBG=IDP    1892    A
DBG=IDP                                                                           DBG=IDP    1893    A
DBG=IDP  *         HERE TO SEARCH FOR STATEMENT KEYWORD.                          DBG=IDP    1894    A
DBG=IDP                                                                           DBG=IDP    1895    A
DBG=IDP   IDP12    MX0    7*CHAR                                                  DBG=IDP    1896    A
DBG=IDP            SX6    B1                                                      DBG=IDP    1897    A
DBG=IDP            ZR     X5,IDP=MN   IF A NULL STMT, IGNORE...                   DBG=IDP    1898    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *                          (PROBABLE "EOS" "EOS" SYNTAX)                DBG=IDP    1899    A
DBG=IDP            SA2    =4040404040404040404040B                                DBG=IDP    1900    A
DBG=IDP            BX1    X0*X5       (X1) = 0LKEYWORD                           DBG=IDP    1901    A
DBG=IDP            IX6    X1-X6       BORROW RIPPLES LEFT TO 1ST NON-ZERO BIT    DBG=IDP    1902    A
DBG=IDP            BX7    -X6+X1      (X7) = ALL TRAILING 0 BITS IN (X1) = 0 BITS DBG=IDP   1903    A
DBG=IDP  *                                  ALL OTHER LEADING BITS = 1 BITS     DBG=IDP    1904    A
DBG=IDP            SB7    60-5        (B7) = RIGHT CIRCULAR SHIFT 5 BITS         DBG=IDP    1905    A
DBG=IDP            BX6    X2*X7       (X6) = 40B WHERE EACH CHAR IN KEYWORD IS   DBG=IDP    1906    A
DBG=IDP            LX7    X6,B7       (X7) = 01B WHERE EACH CHAR IN KEYWORD IS   DBG=IDP    1907    A
DBG=IDP            SB6    IDPKEY                                                 DBG=IDP    1908    A
DBG=IDP            IX7    X6-X7       (X7) = 37B WHERE EACH CHAR IN KEYWORD IS   DBG=IDP    1909    A
DBG=IDP            BX0    X6+X7       (X0) = 77B WHERE EACH CHAR IN KEYWORD IS   DBG=IDP    1910    A
DBG=IDP            RJ     SKT         SEARCH *IDP* KEYWORD TABLE                 DBG=IDP    1911    A
DBG=IDP                                                                          DBG=IDP    1912    A
DBG=IDP   #UKT     IF     DEF,UKT=                                               DBG=IDP    1913    A
DBG=IDP            NZ     X2,IDP=UK   IF A FIND                                  DBG=IDP    1914    A
DBG=IDP            SB6    =XUKT=      (B6) = FWA OF USER KEYWORD TABLE           DBG=IDP    1915    A
DBG=IDP            RJ     SKT         SEARCH USER KEYWORD TABLE                  DBG=IDP    1916    A
DBG=IDP                                                                          DBG=IDP    1917    A
DBG=IDP   IDP=UK   BSS    0                                                      DBG=IDP    1918    A
DBG=IDP   #UKT     ENDIF                                                          DBG=IDP    1919    A
DBG=IDP                                                                          DBG=IDP    1920    A
DBG=IDP            ZR     X2,IDP.ER   IF NO FIND IN TABLE                        DBG=IDP    1921    A
DBG=IDP            SB5    X2          (B5) = PROCESSOR ADDR                      DBG=IDP    1922    A
DBG=IDP            SA5    A5+B1       (A5,X5) = A+C OF TOKEN THAT TERMINATED KEYW DBG=IDP   1923    A
DBG=IDP            JP     B5                                                     DBG=IDP    1924    A
DBG=IDP   IDP=ER   SPACE  4,8                                                    DBG=IDP    1925    A
DBG=IDP  *         HERE FOR ERROR CONDITION.                                     DBG=IDP    1926    A
DBG=IDP                                                                          DBG=IDP    1927    A
DBG=IDP   IDP=ER   BSS    0           ** ERROR **                                DBG=IDP    1928    A
DBG=IDP   IDP.ER   BSS    0                                                      DBG=IDP    1929    A
DBG=IDP                                                                          DBG=IDP    1930    A
DBG=IDP            DUP    L.PRB,1                                                DBG=IDP    1931    A
DBG=IDP            WRITEC =XF.IDO,(=C=  =)                                       DBG=IDP    1932    A
DBG=IDP                                                                          DBG=IDP    1933    A
DBG=IDP            WRITEC =XF.IDO,(=C= ** ERROR **=)                             DBG=IDP    1934    A
DBG=IDP            EQ     IDP=MN                                                 DBG=IDP    1935    A
DBG=IDP   IDP=OV   SPACE  4,8                                                    DBG=IDP    1936    A
DBG=IDP  *         HERE IF TABLE OVERFLOW.                                       DBG=IDP    1937    A
DBG=IDP                                                                          DBG=IDP    1938    A
DBG=IDP   IDP=OV   BSS    0           ** TABLE OVERFLOW **                       DBG=IDP    1939    A
DBG=IDP                                                                          DBG=IDP    1940    A
DBG=IDP            DUP    L.PRB,1                                                DBG=IDP    1941    A
DBG=IDP            WRITEC =XF.IDO,(=C=  =)                                       DBG=IDP    1942    A
DBG=IDP                                                                          DBG=IDP    1943    A
DBG=IDP            WRITEC =XF.IDO,(=C= ** TABLE OVERFLOW **=)                    DBG=IDP    1944    A
DBG=IDP            EQ     IDP=ER                                                 DBG=IDP    1945    A
DBG=IDP   IDP=IFR  SPACE  4,8                                          000280    DBG=IDP    1946    A
DBG=IDP  **        IDP=IFR - IDP RESTART.                              000290    DBG=IDP    1947    A
DBG=IDP  *                                                             000300    DBG=IDP    1948    A
DBG=IDP  *         HERE FROM *RHH* WHEN HOST HAS JUST BEEN RESTARTED   000310    DBG=IDP    1949    A
DBG=IDP  *         VIA *IFR*.  NEED TO --                              000320    DBG=IDP    1950    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *          1.*MEM* BACK DOWN TO ORIGINAL FIELD LENGTH.                 000330  DBG=IDP  1951  A
DBG=IDP  *          2. RESTORE THE WORDS AT (RA.SSW), (FL-2), AND (FL-1).       000340  DBG=IDP  1952  A
DBG=IDP  *          3. (RE)OPEN AND CONNECT INTERACTIVE FILES, JUST IN CASE     000350  DBG=IDP  1953  A
DBG=IDP  *             USER IS RESTARTING ON ANOTHER DAY.                       000360  DBG=IDP  1954  A
DBG=IDP                                                                         000370  DBG=IDP  1955  A
DBG=IDP                                                                         000380  DBG=IDP  1956  A
DBG=IDP   IDP=IFR  BSS    0           ** ENTRY **                               000390  DBG=IDP  1957  A
DBG=IDP            SA4    MEM#HOST                                              000400  DBG=IDP  1958  A
DBG=IDP            SX3    B1                                                    000410  DBG=IDP  1959  A
DBG=IDP            BX6    -X3*X4      CLEAR COMPLETE BIT                        000420  DBG=IDP  1960  A
DBG=IDP            SA6    A4                                                    000430  DBG=IDP  1961  A
DBG=IDP            MEMORY SCM,A4,RCL                                            000440  DBG=IDP  1962  A
DBG=IDP                                                                         000450  DBG=IDP  1963  A
DBG=IDP            SA1    FRZ#SAV                                               000460  DBG=IDP  1964  A
DBG=IDP            SA2    A1+B1                                                 000470  DBG=IDP  1965  A
DBG=IDP            SA3    A2+B1                                                 000480  DBG=IDP  1966  A
DBG=IDP            LX6    X1                                                    000490  DBG=IDP  1967  A
DBG=IDP            BX7    X2                                                    000500  DBG=IDP  1968  A
DBG=IDP            AX4    30                                                    000505  DBG=IDP  1969  A
DBG=IDP            =A6    RA.SSW      RESTORE (RA.SSW)                          000510  DBG=IDP  1970  A
DBG=IDP            SA7    X4-2        RESTORE (FL-2)                            000520  DBG=IDP  1971  A
DBG=IDP            BX6    X3                                                    000530  DBG=IDP  1972  A
DBG=IDP            SA6    A7+B1       RESTORE (FL-1)                            000540  DBG=IDP  1973  A
DBG=IDP            RJ     IIF         INITIALIZE INTERACTIVE FILES              000550  DBG=IDP  1974  A
DBG=IDP            EQ     IDP=MN      CONTINUE INTERACTIVE SESSION...           000560  DBG=IDP  1975  A
DBG=IDP   ST=      SPACE  4,8                                                           DBG=IDP  1976  A
DBG=IDP  **        ST= - STATEMENT PROCESSORS.                                          DBG=IDP  1977  A
DBG=IDP  *                                                                              DBG=IDP  1978  A
DBG=IDP   ST=BRK   SPACE  4,8                                                           DBG=IDP  1979  A
DBG=IDP  *         HERE TO PROCESS *BREAK ADDR,LL,UL,INC*,                              DBG=IDP  1980  A
DBG=IDP  *                    OR *BRPL ADDR,ADDR1,LL,UL,INC* FORM,                      DBG=IDP  1981  A
DBG=IDP  *                    OR  *BREQ ADDR,ADDR1,ADDR2,LL,UL,INC* FORM.               DBG=IDP  1982  A
DBG=IDP                                                                                 DBG=IDP  1983  A
DBG=IDP   ST=BRK   SX6    B7+IDPKEY-IDPKBR  (X6) = BREAK TYPE CODE (BTC)                DBG=IDP  1984  A
DBG=IDP            SA6    IDPBTC                                                       DBG=IDP  1985  A
DBG=IDP            ZR     X5,ST=BRK13 IF EOS ENCOUNTERED, LIST ALL BREAKS...           DBG=IDP  1986  A
DBG=IDP            RJ     PAS         PARSE SUBEXPRESSION-- ADDR                       DBG=IDP  1987  A
DBG=IDP            ZR     X3,ST=BRK13 IF 1ST EXPRESSION IS NULL, LIST ALL BREAKS       DBG=IDP  1988  A
DBG=IDP            SX6    X1                                                           DBG=IDP  1989  A
DBG=IDP            SB2    X1          (B2) = ADDR AT WHICH TO BREAK                    DBG=IDP  1990  A
DBG=IDP            SA6    AP=FWA                                                       DBG=IDP  1991  A
DBG=IDP            RJ     CHK         CHECK CM ADDRESS                                 DBG=IDP  1992  A
DBG=IDP            MI     B2,IDP.ER   IF ADDR IS BAD                                   DBG=IDP  1993  A
DBG=IDP            SA2    IDPBTC                                                       DBG=IDP  1994  A
DBG=IDP            MX3    0                                                            DBG=IDP  1995  A
DBG=IDP            ZR     X2,ST=BRK2  IF THIS IS *BREAK ADDR,LL,UL,INC*                DBG=IDP  1996  A
DBG=IDP                                                                                 DBG=IDP  1997  A
DBG=IDP  *         ASSEMBLE ADDR1.                                                      DBG=IDP  1998  A
DBG=IDP                                                                                 DBG=IDP  1999  A
DBG=IDP            ZR     X5,ST=BRK2  IF ADDR FOLLOWED BY EOS                          DBG=IDP  2000  A
DBG=IDP            RJ     PAS         PARSE SUBEXPRESSION-- ADDR1                      DBG=IDP  2001  A
DBG=IDP                                                                                 DBG=IDP  2002  A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP   ST=BRK2  NZ     X3,ST=BRK3  IF ADDR1 NOT NULL                          DBG=IDP    2003    A
DBG=IDP            SX1    0           (X1) = DEFAULT ADDR1                       DBG=IDP    2004    A
DBG=IDP                                                                          DBG=IDP    2005    A
DBG=IDP   ST=BRK3  SB2    X1                                                     DBG=IDP    2006    A
DBG=IDP            RJ     CHK         CHECK CM ADDRESS                           DBG=IDP    2007    A
DBG=IDP            MI     B2,IDP.ER   IF ADDR1 IS BAD                            DBG=IDP    2008    A
DBG=IDP            SX6    B2                                                     DBG=IDP    2009    A
DBG=IDP            MX3    0                                                      DBG=IDP    2010    A
DBG=IDP            SA6    AP=LWA                                                 DBG=IDP    2011    A
DBG=IDP            SA2    IDPBTC                                                 DBG=IDP    2012    A
DBG=IDP            SX7    X2-BTC=EQ                                              DBG=IDP    2013    A
DBG=IDP            MI     X7,ST=BRK4  IF THIS IS *BREAK ADDR,LL,UL,INC*          DBG=IDP    2014    A
DBG=IDP  *                               OR *BRPL ADDR,ADDR1,LL,UL,INC* FORM   DBG=IDP    2015    A
DBG=IDP                                                                          DBG=IDP    2016    A
DBG=IDP  *         ASSEMBL ADDR2.                                                DBG=IDP    2017    A
DBG=IDP                                                                          DBG=IDP    2018    A
DBG=IDP            ZR     X5,ST=BRK4  IF ADDR1 FOLLOWED BY EOS                   DBG=IDP    2019    A
DBG=IDP            RJ     PAS         PARSE SUBEXPRESSION-- ADDR2                DBG=IDP    2020    A
DBG=IDP                                                                          DBG=IDP    2021    A
DBG=IDP   ST=BRK4  NZ     X3,ST=BRK5  IF ADDR2 NOT NULL                          DBG=IDP    2022    A
DBG=IDP            SX1    0                                                      DBG=IDP    2023    A
DBG=IDP                                                                          DBG=IDP    2024    A
DBG=IDP   ST=BRK5  SB2    X1                                                     DBG=IDP    2025    A
DBG=IDP            RJ     CHK         CHECK CM ADDRESS                           DBG=IDP    2026    A
DBG=IDP            MI     B2,IDP.ER   IF ADDR2 IS BAD                            DBG=IDP    2027    A
DBG=IDP            SX6    B2                                                     DBG=IDP    2028    A
DBG=IDP            MX3    0                                                      DBG=IDP    2029    A
DBG=IDP            SA6    AP=LEN      SAVE (X6) = ADDR2                          DBG=IDP    2030    A
DBG=IDP                                                                          DBG=IDP    2031    A
DBG=IDP  *         ASSEMBLE LL.                                                  DBG=IDP    2032    A
DBG=IDP                                                                          DBG=IDP    2033    A
DBG=IDP            ZR     X5,ST=BRK6  IF ADDR2 FOLLOWED BY EOS                   DBG=IDP    2034    A
DBG=IDP            RJ     PAS         PARSE SUBEXPRESSION-- LL                   DBG=IDP    2035    A
DBG=IDP                                                                          DBG=IDP    2036    A
DBG=IDP   ST=BRK6  NZ     X3,ST=BRK7  IF LL NOT NULL                             DBG=IDP    2037    A
DBG=IDP            SX1    1           (X1) = DEFAULT LL                          DBG=IDP    2038    A
DBG=IDP                                                                          DBG=IDP    2039    A
DBG=IDP   ST=BRK7  SX6    X1                                                     DBG=IDP    2040    A
DBG=IDP            SA6    AP=LL                                                  DBG=IDP    2041    A
DBG=IDP            MX3    0                                                      DBG=IDP    2042    A
DBG=IDP            ZR     X5,ST=BRK8  IF LL FOLLOWED BY EOS                      DBG=IDP    2043    A
DBG=IDP            RJ     PAS         PARSE SUBEXPRESSION-- UL                   DBG=IDP    2044    A
DBG=IDP                                                                          DBG=IDP    2045    A
DBG=IDP   ST=BRK8  NZ     X3,ST=BRK9  IF UL NOT NULL                             DBG=IDP    2046    A
DBG=IDP            SX1    100         (X1) = DEFAULT UL                          DBG=IDP    2047    A
DBG=IDP                                                                          DBG=IDP    2048    A
DBG=IDP   ST=BRK9  SX6    X1                                                     DBG=IDP    2049    A
DBG=IDP            SA6    AP=UL                                                  DBG=IDP    2050    A
DBG=IDP            MX3    0                                                      DBG=IDP    2051    A
DBG=IDP            ZR     X5,ST=BRK10 IF UL FOLLOWED BY EOS                      DBG=IDP    2052    A
DBG=IDP            RJ     PAS         PARSE SUBEXPRESSION-- INC                  DBG=IDP    2053    A
DBG=IDP                                                                          DBG=IDP    2054    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP   ST=BRK10 NZ     X3,ST=BRK11 IF INC NOT NULL                          DBG=IDP     2055     A
DBG=IDP            SX1    1           (X1) = DEFAULT INC                       DBG=IDP     2056     A
DBG=IDP                                                                        DBG=IDP     2057     A
DBG=IDP   ST=BRK11 SX6    X1                                                   DBG=IDP     2058     A
DBG=IDP            SA6    AP=INC                                               DBG=IDP     2059     A
DBG=IDP                                                                        DBG=IDP     2060     A
DBG=IDP   *        ASSEMBLE NEW BREAK TABLE ENTRY.                             DBG=IDP     2061     A
DBG=IDP                                                                        DBG=IDP     2062     A
DBG=IDP            SA1    AP=FWA      (X1) = BREAK ADDR                        DBG=IDP     2063     A
DBG=IDP            SA2    A1+B1       (X2) = ADDR1                             DBG=IDP     2064     A
DBG=IDP            SA3    A2+B1       (X3) = ADDR2                             DBG=IDP     2065     A
DBG=IDP            SA4    IDPBTC      (X4) = BREAK TYPE CODE                   DBG=IDP     2066     A
DBG=IDP            LX2    54-18                                                DBG=IDP     2067     A
DBG=IDP            BX6    X1+X2       (X6) = 6/0,18/ADDR1,18/0,18/BREAK ADDR   DBG=IDP     2068     A
DBG=IDP            LX3    36-18                                                DBG=IDP     2069     A
DBG=IDP            BX6    X6+X3       (X6) = 6/0,18/ADDR1,18/ADDR2,18/BRK ADDR DBG=IDP     2070     A
DBG=IDP            LX4    60-6                                                 DBG=IDP     2071     A
DBG=IDP            BX6    X6+X4       (X6) = 6/BTC,18/ADDR1,18/ADDR2,18/BRK ADDR DBG=IDP   2072     A
DBG=IDP            SA6    ST=BRKB                                              DBG=IDP     2073     A
DBG=IDP                                                                        DBG=IDP     2074     A
DBG=IDP            SA2    AP=LL       (X2) = LL                                DBG=IDP     2075     A
DBG=IDP            SA3    A2+B1       (X3) = UL                                DBG=IDP     2076     A
DBG=IDP            SA4    A3+B1       (X4) = INC                               DBG=IDP     2077     A
DBG=IDP            LX2    60-15                                                DBG=IDP     2078     A
DBG=IDP            LX3    45-15                                                DBG=IDP     2079     A
DBG=IDP            BX7    X2+X3       (X7) = 15/LL,15/UL,12/0,18/0             DBG=IDP     2080     A
DBG=IDP            LX4    30-12                                                DBG=IDP     2081     A
DBG=IDP            BX7    X7+X4       (X7) = 15/LL,15/UL,12/INC,18/0           DBG=IDP     2082     A
DBG=IDP            SA7    A6+B1                                                DBG=IDP     2083     A
DBG=IDP                                                                        DBG=IDP     2084     A
DBG=IDP            SX1    X1                                                   DBG=IDP     2085     A
DBG=IDP            SB6    IDPBA                                                DBG=IDP     2086     A
DBG=IDP            MX0    -18                                                  DBG=IDP     2087     A
DBG=IDP            BX0    -X0                                                  DBG=IDP     2088     A
DBG=IDP            RJ     SKT         SEARCH FOR BREAK ADDR                    DBG=IDP     2089     A
DBG=IDP            SA1    ST=BRKB                                              DBG=IDP     2090     A
DBG=IDP            NZ     X2,ST=BRK12 IF BREAK ADDR ALREADY IN TABLE           DBG=IDP     2091     A
DBG=IDP                                                                        DBG=IDP     2092     A
DBG=IDP   *        HERE IF MAKING A NEW ENTRY.                                 DBG=IDP     2093     A
DBG=IDP                                                                        DBG=IDP     2094     A
DBG=IDP            BX6    X1                                                   DBG=IDP     2095     A
DBG=IDP            RJ     ADZ         ADD NEW *IDPBA* ENTRY                    DBG=IDP     2096     A
DBG=IDP            SA3    X1          (X3) = PREVIOUS CONTENTS OF BREAK ADDR   DBG=IDP     2097     A
DBG=IDP            SA4    ST=BRKA     (X4) = *RJ IDP* PLUG                     DBG=IDP     2098     A
DBG=IDP            BX6    X3                                                   DBG=IDP     2099     A
DBG=IDP            LX7    X4                                                   DBG=IDP     2100     A
DBG=IDP            SB6    B7+B7                                                DBG=IDP     2101     A
DBG=IDP            SA6    IDPBC+B6                                             DBG=IDP     2102     A
DBG=IDP            SA7    A3                                                   DBG=IDP     2103     A
DBG=IDP            SA1    ST=BRKB+1   (X1) = 2ND WORD OF NEW *IDPBC* ENTRY     DBG=IDP     2104     A
DBG=IDP            BX6    X1                                                   DBG=IDP     2105     A
DBG=IDP            SA6    A6+B1                                                DBG=IDP     2106     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            EQ     IDP=MN                                                  DBG=IDP     2107     A
DBG=IDP                                                                            DBG=IDP     2108     A
DBG=IDP  *         HERE IF REPLACING/UPDATING AN EXISTING ENTRY.                   DBG=IDP     2109     A
DBG=IDP                                                                            DBG=IDP     2110     A
DBG=IDP   ST=BRK12 SB6    B7+B7                                                    DBG=IDP     2111     A
DBG=IDP            BX6    X1                                                       DBG=IDP     2112     A
DBG=IDP            SA3    IDPBC+B6+1  (X3) = 2ND WORD OF EXISTING *IDPBC* ENTRY    DBG=IDP     2113     A
DBG=IDP            SA6    A2          REPLACE OLD *IDPBA* WITH NEW                 DBG=IDP     2114     A
DBG=IDP            SA4    A1+B1       (X4) = NEW 2ND WORD OF *IDPBC* ENTRY         DBG=IDP     2115     A
DBG=IDP            SX3    X3          (X3) = SNAP COUNT                            DBG=IDP     2116     A
DBG=IDP            BX6    X4+X3                                                    DBG=IDP     2117     A
DBG=IDP            SA6    A3          REPLACE OLD WITH NEW                         DBG=IDP     2118     A
DBG=IDP            EQ     IDP=MN                                                   DBG=IDP     2119     A
DBG=IDP                                                                            DBG=IDP     2120     A
DBG=IDP  *         HERE TO LIST ALL BREAKS.                                        DBG=IDP     2121     A
DBG=IDP                                                                            DBG=IDP     2122     A
DBG=IDP   ST=BRK13 RJ     LBT         LIST BREAK TABLE                             DBG=IDP     2123     A
DBG=IDP            EQ     IDP=MN                                                   DBG=IDP     2124     A
DBG=IDP                                                                            DBG=IDP     2125     A
DBG=IDP                                                                            DBG=IDP     2126     A
DBG=IDP   ST=BRKA  RJ     IDP=        PLUG VALUE                                   DBG=IDP     2127     A
DBG=IDP   -        VFD    30/0                                                     DBG=IDP     2128     A
DBG=IDP                                                                            DBG=IDP     2129     A
DBG=IDP   ST=BRKB  BSSZ   2           SAVE CELLS                                   DBG=IDP     2130     A
DBG=IDP   ST=COD   SPACE  4,8                                                      DBG=IDP     2131     A
DBG=IDP  *         HERE TO PROCESS *CODE FWA,LWA,LEN*.                             DBG=IDP     2132     A
DBG=IDP                                                                            DBG=IDP     2133     A
DBG=IDP   ST=COD   BSS    0                                                        DBG=IDP     2134     A
DBG=IDP            SA1    IDPPREG                                        000300    DBG=IDP     2135     A
DBG=IDP            SX6    B1          (X6) = DEFAULT LEN                 000310    DBG=IDP     2136     A
DBG=IDP            BX7    X1          (X7) = DEFAULT FWA                 000320    DBG=IDP     2137     A
DBG=IDP            SA6    PAT#LEN                                        000330    DBG=IDP     2138     A
DBG=IDP            SA7    PAT#FWA                                        000340    DBG=IDP     2139     A
DBG=IDP            RJ     PAT         PARSE *FWA,LWA,LEN* TRIPLE         000350    DBG=IDP     2140     A
DBG=IDP            MI     B2,IDP=ER   IF FWA, LWA, OR LEN IS BAD         000360    DBG=IDP     2141     A
DBG=IDP                                                                            DBG=IDP     2142     A
DBG=IDP  *         HERE TO DEASSEMBLE FROM FWA TO LWA.                             DBG=IDP     2143     A
DBG=IDP                                                                            DBG=IDP     2144     A
DBG=IDP   ST=COD5  SA1    AP=FWA                                                   DBG=IDP     2145     A
DBG=IDP            SA2    A1+B1                                                    DBG=IDP     2146     A
DBG=IDP            IX3    X2-X1                                                    DBG=IDP     2147     A
DBG=IDP            SX6    X1+B1                                                    DBG=IDP     2148     A
DBG=IDP            MI     X3,IDP=MN   IF FINISHED                                  DBG=IDP     2149     A
DBG=IDP            SA6    A1                                                       DBG=IDP     2150     A
DBG=IDP            SA5    X1          (A5,X5) = WORD TO DEASSEMBLE                 DBG=IDP     2151     A
DBG=IDP            SB4    60          (B4) = POSITION COUNTER                      DBG=IDP     2152     A
DBG=IDP                                                                            DBG=IDP     2153     A
DBG=IDP   ST=COD6  RJ     DAZ         DEASSEMBLE INSTRUCTION                       DBG=IDP     2154     A
DBG=IDP            MI     X1,ST=COD5  IF INSTRUCTION FORCED UPPER                  DBG=IDP     2155     A
DBG=IDP            SB4    B4-B5                                                    DBG=IDP     2156     A
DBG=IDP            GT     B4,B0,ST=COD6  IF MORE INSTRUCTIONS TO GO IN THIS WORD   DBG=IDP     2157     A
DBG=IDP            EQ     ST=COD5                                                  DBG=IDP     2158     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP   ST=CON   SPACE  4,8                                                  DBG=IDP    2159    A
DBG=IDP  *         HERE TO PROCESS *CONNECT FILE*.                            DBG=IDP    2160    A
DBG=IDP                                                                       DBG=IDP    2161    A
DBG=IDP   ST=CON   ZR     X5,IDP=ER   IF EOS ENCOUTNTERED                     DBG=IDP    2162    A
DBG=IDP            SA5    A5+B1                                               DBG=IDP    2163    A
DBG=IDP            SB4    B0          SET TO *CONNECT*                        DBG=IDP    2164    A
DBG=IDP                                                                       DBG=IDP    2165    A
DBG=IDP   ST=CON2  BSS    0                                                   DBG=IDP    2166    A
DBG=IDP                                                                       DBG=IDP    2167    A
DBG=IDP   #UFT     IF     DEF,UFT=                                            DBG=IDP    2168    A
DBG=IDP            SB6    =XUFT=      (B6) = FWA OF USER FILE NAME TABLE      DBG=IDP    2169    A
DBG=IDP            MX0    7*CHAR                                              DBG=IDP    2170    A
DBG=IDP            BX1    X0*X5       (X1) = 0LLFN                            DBG=IDP    2171    A
DBG=IDP            RJ     SKT         SEARCH FOR FILE NAME                    DBG=IDP    2172    A
DBG=IDP            ZR     X2,IDP=ER   IF NO FIND                              DBG=IDP    2173    A
DBG=IDP            SA1    X2          (A1,X1) = A + C OF 1ST WORD OF FET      DBG=IDP    2174    A
DBG=IDP            SX2    B4          SET TO *CONNECT/DISCONT*                DBG=IDP    2175    A
DBG=IDP            RJ     CON         CONNECT/DISCONT FILE                    DBG=IDP    2176    A
DBG=IDP                                                                       DBG=IDP    2177    A
DBG=IDP   #OS3     IFEQ   .OS,3       IF SCOPE 3                              DBG=IDP    2178    A
DBG=IDP            MI     X1,IDP=ER   IF *CON* DETECTED ERROR...              DBG=IDP    2179    A
DBG=IDP   #OS3     ENDIF                                                      DBG=IDP    2180    A
DBG=IDP   #UFT     ENDIF                                                      DBG=IDP    2181    A
DBG=IDP                                                                       DBG=IDP    2182    A
DBG=IDP            EQ     IDP=MN                                              DBG=IDP    2183    A
DBG=IDP   ST=DPC   SPACE  4,8                                                 DBG=IDP    2184    A
DBG=IDP  *         HERE TO PROCESS *DPC FWA,LWA,LEN*.                         DBG=IDP    2185    A
DBG=IDP                                                                       DBG=IDP    2186    A
DBG=IDP   ST=DPC   BSS    0                                                   DBG=IDP    2187    A
DBG=IDP            MX6    -1          (X6) = DEFAULT FWA  (ERROR)       000380  DBG=IDP    2188    A
DBG=IDP            SX7    B1          (X7) = DEFAULT LEN                000390  DBG=IDP    2189    A
DBG=IDP            SA6    PAT#FWA                                       000400  DBG=IDP    2190    A
DBG=IDP            SA7    PAT#LEN                                       000410  DBG=IDP    2191    A
DBG=IDP            RJ     PAT         PARSE *FWA,LWA,LEN* TRIPLE        000420  DBG=IDP    2192    A
DBG=IDP            MI     B2,IDP=ER   IF FWA, LWA, OR LEN IS BAD        000430  DBG=IDP    2193    A
DBG=IDP            SA2    AP=FWA      (X2) = FWA                        000440  DBG=IDP    2194    A
DBG=IDP            SA3    A2+B1       (X3) = LWA                        000450  DBG=IDP    2195    A
DBG=IDP            BX4    X4-X4       (X4) = LEN (I.E. NO LEN)          000460  DBG=IDP    2196    A
DBG=IDP            RJ     DOD         DUMP CENTRAL MEMORY - OCTAL AND DPC       DBG=IDP    2197    A
DBG=IDP            PL     B2,IDP=MN   IF FWA,LWA,AND LEN OK                     DBG=IDP    2198    A
DBG=IDP            EQ     IDP=ER                                             DBG=IDP    2199    A
DBG=IDP   ST=DSC   SPACE  4,8                                                 DBG=IDP    2200    A
DBG=IDP  *         HERE TO PROCESS *DISCONT FILE*.                            DBG=IDP    2201    A
DBG=IDP                                                                       DBG=IDP    2202    A
DBG=IDP   ST=DSC   ZR     X5,IDP=ER   IF EOS ENCOUNTERED                      DBG=IDP    2203    A
DBG=IDP            SA5    A5+B1                                               DBG=IDP    2204    A
DBG=IDP            SB4    B1          SET TO *DISCONT*                        DBG=IDP    2205    A
DBG=IDP            EQ     ST=CON2     PROCESS *DISCONT*...                    DBG=IDP    2206    A
DBG=IDP   ST=END   SPACE  4,8                                                 DBG=IDP    2207    A
DBG=IDP  *         HERE TO PROCESS *END*.                                     DBG=IDP    2208    A
DBG=IDP                                                                       DBG=IDP    2209    A
DBG=IDP   ST=END   BSS    0           ** END **                               DBG=IDP    2210    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            PRIDP  (=C= BYE...=),,L.PRB                                  DBG=IDP    2211    A
DBG=IDP            WRITER =XF.IDO,,RCL                                          DBG=IDP    2212    A
DBG=IDP                                                                         DBG=IDP    2213    A
DBG=IDP   ST=ENDW  SA1    IDPPREG     (X1) = PSEUDO P REGISTER                  DBG=IDP    2214    A
DBG=IDP            SA2    IDPPOS      (X2) = POS COUNTER                        DBG=IDP    2215    A
DBG=IDP            SA3    ST=ENDX                                               DBG=IDP    2216    A
DBG=IDP            SA4    =46000460004600046000B                                DBG=IDP    2217    A
DBG=IDP            SB4    X2                                                    DBG=IDP    2218    A
DBG=IDP            LX6    X3                                                    DBG=IDP    2219    A
DBG=IDP            SB6    60                                                    DBG=IDP    2220    A
DBG=IDP            SA5    ST=ENDZ     (X5) = *JP B0+0* PLUG                     DBG=IDP    2221    A
DBG=IDP            LE     B4,B0,*+4S15  IF BAD POS COUNTER, SHAZAM...           DBG=IDP    2222    A
DBG=IDP            LT     B4,B6,ST=END2 IF NEXT INSTRUCTION TO XEQ IS NOT UPPER, DBG=IDP    2223    A
DBG=IDP  *                              (I.E.STEP MODE LEFT US IN MIDDLE OF WORD) DBG=IDP    2224    A
DBG=IDP            NZ     X3,ST=END4  IF TO XEQ PREVIOUS CONTENTS OF BREAK ADDR DBG=IDP    2225    A
DBG=IDP  *                              AT *ST=ENDX*                          DBG=IDP    2226    A
DBG=IDP            LX6    X4                                                    DBG=IDP    2227    A
DBG=IDP            EQ     ST=END4                                               DBG=IDP    2228    A
DBG=IDP                                                                         DBG=IDP    2229    A
DBG=IDP   ST=END2  NZ     X3,ST=END3  IF TO XEQ PARTIAL WORD AT *ST=ENDX*       DBG=IDP    2230    A
DBG=IDP            SA3    X1          (X3) = PARTIALLY XEQ WORD AT ((IDPPREG))  DBG=IDP    2231    A
DBG=IDP            SX1    X1+B1       (X1) = RETURN ADDR                        DBG=IDP    2232    A
DBG=IDP                                                                         DBG=IDP    2233    A
DBG=IDP  *         HERE BECAUSE STEP MODE LEFT US IN THE MIDDLE OF A WORD--     DBG=IDP    2234    A
DBG=IDP  *           LEFT JUSTIFY REMAINING INSTRUCTION(S) TO XEQ AND NO-OP FILL. DBG=IDP    2235    A
DBG=IDP                                                                         DBG=IDP    2236    A
DBG=IDP   ST=END3  MX0    1                                                     DBG=IDP    2237    A
DBG=IDP            SB7    B6-B4       (B7) = NR OF BITS ALREADY XEQ IN THIS WORD DBG=IDP    2238    A
DBG=IDP            SB5    B4-B1                                                 DBG=IDP    2239    A
DBG=IDP            LX6    X3,B7       (X6) = INSTRUCTIONS TO XEQ-- LEFT JUSTIFIED DBG=IDP    2240    A
DBG=IDP            AX0    X0,B5       (X0) = EXTRACT MASK FOR INST TO XEQ       DBG=IDP    2241    A
DBG=IDP            BX6    X0*X6                                                 DBG=IDP    2242    A
DBG=IDP            BX4    -X0*X4                                                DBG=IDP    2243    A
DBG=IDP            BX6    X6+X4       (X6) = INSTRUCTIONS TO XEQ W/ NO-OP FILL  DBG=IDP    2244    A
DBG=IDP                                                                         DBG=IDP    2245    A
DBG=IDP  *         HERE TO SET UP *JP B0+RETURN ADDR*.                          DBG=IDP    2246    A
DBG=IDP                                                                         DBG=IDP    2247    A
DBG=IDP   ST=END4  SA6    ST=ENDX                                               DBG=IDP    2248    A
DBG=IDP            LX1    30                                                    DBG=IDP    2249    A
DBG=IDP            BX6    X5+X1       (X6) = 30/JP B0+RETURN ADDR, 30/NO-OP     DBG=IDP    2250    A
DBG=IDP            SA6    ST=ENDY                                               DBG=IDP    2251    A
DBG=IDP            RJ     RSR=        RESTORE ORIGINAL REGISTERS                DBG=IDP    2252    A
DBG=IDP                                                                         DBG=IDP    2253    A
DBG=IDP   ST=ENDX  BSSZ   1           WORD TO EXECUTE BEFORE EXITING *IDP*      DBG=IDP    2254    A
DBG=IDP  *                              IF AN IDP GENERATED BREAK, THEN IS PLUG- DBG=IDP    2255    A
DBG=IDP  *                              GED WITH PREVIOUS CONTENTS OF BREAK ADDR DBG=IDP    2256    A
DBG=IDP  *                              IF AN ASSEMBLED BREAK, THEN IS .ZR.      DBG=IDP    2257    A
DBG=IDP  *                              SEE *BRK*.                               DBG=IDP    2258    A
DBG=IDP                                                                         DBG=IDP    2259    A
DBG=IDP   ST=ENDY  BSSZ   1           PLUGGED WITH *JP B0+RETURN ADDR*          DBG=IDP    2260    A
DBG=IDP                                                                         DBG=IDP    2261    A
DBG=IDP   ST=ENDZ  JP     B0+         PLUG VALUE                                DBG=IDP    2262    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP   ST=FRZ   SPACE  4,8                                          000580   DBG=IDP     2263     A
DBG=IDP  *         HERE TO PROCESS *FREEZE*.                           000590   DBG=IDP     2264     A
DBG=IDP                                                                000600   DBG=IDP     2265     A
DBG=IDP   ST=FRZ   BSS    0                                            000610   DBG=IDP     2266     A
DBG=IDP            SA1    FW.IFR+/IFR/F.FRZ-RA.ORG                     000620   DBG=IDP     2267     A
DBG=IDP            SA2    =0LFRZ                                       000630   DBG=IDP     2268     A
DBG=IDP            MX3    -18                                          000640   DBG=IDP     2269     A
DBG=IDP            BX1    -X3*X1      CLEAR OLD LFN                    000650   DBG=IDP     2270     A
DBG=IDP            IX6    X1+X2       MERGE NEW LFN                    000660   DBG=IDP     2271     A
DBG=IDP            SA6    A1                                           000670   DBG=IDP     2272     A
DBG=IDP            RJ     FRZ         FREEZE INTERACTIVE HOST          000680   DBG=IDP     2273     A
DBG=IDP            EQ     IDP=MN                                       000690   DBG=IDP     2274     A
DBG=IDP   ST=JP    SPACE  4,8                                                    DBG=IDP     2275     A
DBG=IDP  *         HERE TO PROCESS *JUMP ADDR*.                                  DBG=IDP     2276     A
DBG=IDP                                                                          DBG=IDP     2277     A
DBG=IDP   ST=JP    BSS    0                                                      DBG=IDP     2278     A
DBG=IDP            ZR     X5,IDP=MN   IF EOS ENCOUNTERED                        DBG=IDP     2279     A
DBG=IDP            RJ     PAS         PARSE SUBEXPRESSION-- ADDR                DBG=IDP     2280     A
DBG=IDP            ZR     X3,IDP=MN   IF ADDR IS NULL                           DBG=IDP     2281     A
DBG=IDP            SB2    X1                                                    DBG=IDP     2282     A
DBG=IDP            RJ     CHK         CHECK CM ADDRESS                          DBG=IDP     2283     A
DBG=IDP            MI     B2,IDP.ER   IF ADDR IS BAD                            DBG=IDP     2284     A
DBG=IDP            SA1    =46000460004600046000B                                DBG=IDP     2285     A
DBG=IDP            SX7    60                                                    DBG=IDP     2286     A
DBG=IDP            SA7    IDPPOS                                                DBG=IDP     2287     A
DBG=IDP            SX6    B2                                                    DBG=IDP     2288     A
DBG=IDP            BX7    X1                                                    DBG=IDP     2289     A
DBG=IDP            SA6    IDPPREG                                               DBG=IDP     2290     A
DBG=IDP            SA7    ST=ENDX                                               DBG=IDP     2291     A
DBG=IDP            EQ     ST=END                                                DBG=IDP     2292     A
DBG=IDP   ST=OUT   SPACE  4,8                                                    DBG=IDP     2293     A
DBG=IDP  **        HERE TO PROCESS *OUTPUT I,B*.                                 DBG=IDP     2294     A
DBG=IDP                                                                          DBG=IDP     2295     A
DBG=IDP   ST=OUT   ZR     X5,IDP=MN   IF EOS ENCOUNTERED                        DBG=IDP     2296     A
DBG=IDP            SB6    IDPKOUT     (B6) = FWA OF *OUTPUT* OPTIONS KEYWORD TBL DBG=IDP     2297     A
DBG=IDP            RJ     POL         PROCESS OPTIONS LIST ITEM                 DBG=IDP     2298     A
DBG=IDP            SA1    B6+B7                                                 DBG=IDP     2299     A
DBG=IDP            ZR     X3,ST=OUT   IF NULL PARAMETER, IGNORE...              DBG=IDP     2300     A
DBG=IDP            SB5    X1          (B5) = OPTIONS PROCESSOR ADDR             DBG=IDP     2301     A
DBG=IDP            MX0    1                                                     DBG=IDP     2302     A
DBG=IDP            SA2    IDPFLG                                                DBG=IDP     2303     A
DBG=IDP            BX6    -X6*X0      (X6) = +0 IF -OPTION, OR 1S59 IF +OPTION  DBG=IDP     2304     A
DBG=IDP            JP     B5+                                                   DBG=IDP     2305     A
DBG=IDP                                                                          DBG=IDP     2306     A
DBG=IDP  *         HERE IF *OUTPUT I* OR *OUTPUT -I*.                            DBG=IDP     2307     A
DBG=IDP                                                                          DBG=IDP     2308     A
DBG=IDP   ST=OUT2  LX0    IDF.IDOP-59                                           DBG=IDP     2309     A
DBG=IDP            BX2    -X0*X2      CLEAR EXISTING FIELD                      DBG=IDP     2310     A
DBG=IDP            LX6    IDF.IDOP-59                                           DBG=IDP     2311     A
DBG=IDP            BX6    X2+X6                                                 DBG=IDP     2312     A
DBG=IDP            SA6    A2                                                    DBG=IDP     2313     A
DBG=IDP            EQ     ST=OUT                                                DBG=IDP     2314     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP                                                                    DBG=IDP    2315    A
DBG=IDP  *         HERE IF *OUTPUT B* OR *OUTPUT -B*.                      DBG=IDP    2316    A
DBG=IDP                                                                    DBG=IDP    2317    A
DBG=IDP   ST=OUT3  LX0    IDF.BDOP-59                                      DBG=IDP    2318    A
DBG=IDP            BX2    -X0*X2      CLEAR EXISTING FIELD                 DBG=IDP    2319    A
DBG=IDP            LX6    IDF.BDOP-59                                      DBG=IDP    2320    A
DBG=IDP            BX6    X2+X6                                            DBG=IDP    2321    A
DBG=IDP            SA6    A2                                               DBG=IDP    2322    A
DBG=IDP            EQ     ST=OUT                                           DBG=IDP    2323    A
DBG=IDP                                                                    DBG=IDP    2324    A
DBG=IDP  *         HERE IF *OUTPUT BE* OR *OUTPUT -BE*.                    DBG=IDP    2325    A
DBG=IDP                                                                    DBG=IDP    2326    A
DBG=IDP   ST=OUT4  LX0    IDF.BKOP-59                                      DBG=IDP    2327    A
DBG=IDP            BX2    -X0*X2      CLEAR EXISTING FIELD                 DBG=IDP    2328    A
DBG=IDP            LX6    IDF.BKOP-59                                      DBG=IDP    2329    A
DBG=IDP            BX6    X2+X6       MERGE NEW BKO                        DBG=IDP    2330    A
DBG=IDP            SA6    A2                                               DBG=IDP    2331    A
DBG=IDP            EQ     ST=OUT                                           DBG=IDP    2332    A
DBG=IDP                                                                    DBG=IDP    2333    A
DBG=IDP  *         HERE IF *OUTPUT IE* OR *OUTPUT -IE*.                    DBG=IDP    2334    A
DBG=IDP                                                                    DBG=IDP    2335    A
DBG=IDP   ST=OUT5  LX0    IDF.IKOP-59                                      DBG=IDP    2336    A
DBG=IDP            BX2    -X0*X2      CLEAR EXISTING FIELD                 DBG=IDP    2337    A
DBG=IDP            LX6    IDF.IKOP-59                                      DBG=IDP    2338    A
DBG=IDP            BX6    X2+X6       MERGE NEW IKO                        DBG=IDP    2339    A
DBG=IDP            SA6    A2                                               DBG=IDP    2340    A
DBG=IDP            EQ     ST=OUT                                           DBG=IDP    2341    A
DBG=IDP   ST=REG   SPACE  4,8                                              DBG=IDP    2342    A
DBG=IDP  *         HERE TO PROCESS *REG R1,R2,...,RN*.                     DBG=IDP    2343    A
DBG=IDP                                                                    DBG=IDP    2344    A
DBG=IDP   ST=REG   BSS    0                                                DBG=IDP    2345    A
DBG=IDP            SA0    A5          INITIALIZE (A0) = ADDR OF CURRENT TOKEN DBG=IDP 2346    A
DBG=IDP            NZ     X5,ST=REG2  IF *REG* NOT FOLLOWED BY EOS         DBG=IDP    2347    A
DBG=IDP            RJ     DAR         DUMP ALL REGISTERS                   DBG=IDP    2348    A
DBG=IDP            EQ     IDP=MN                                           DBG=IDP    2349    A
DBG=IDP                                                                    DBG=IDP    2350    A
DBG=IDP  *         HERE FOR SELECTED REGISTER DUMP.                        DBG=IDP    2351    A
DBG=IDP                                                                    DBG=IDP    2352    A
DBG=IDP   ST=REG2  SA0    A0+B1                                            DBG=IDP    2353    A
DBG=IDP            SA5    A0                                               DBG=IDP    2354    A
DBG=IDP            ZR     X5,IDP=MN   IF AN EOS ENCOUNTERED                DBG=IDP    2355    A
DBG=IDP                                                                    DBG=IDP    2356    A
DBG=IDP            SX6    X5-O.SEP                                         DBG=IDP    2357    A
DBG=IDP            SX7    X5-O.CONS                                        DBG=IDP    2358    A
DBG=IDP            SA4    IDP0TR                                           DBG=IDP    2359    A
DBG=IDP            ZR     X6,ST=REG2  IF SEPARATOR TOKEN                   DBG=IDP    2360    A
DBG=IDP            ZR     X7,ST=REG4  IF CONSTANT TOKEN                    DBG=IDP    2361    A
DBG=IDP                                                                    DBG=IDP    2362    A
DBG=IDP  *         HERE IF A NAME TOKEN, CHECK FOR REGISTER NAME.          DBG=IDP    2363    A
DBG=IDP                                                                    DBG=IDP    2364    A
DBG=IDP            LX5    CHAR                                             DBG=IDP    2365    A
DBG=IDP            MX0    -CHAR                                            DBG=IDP    2366    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            BX6    -X0*X5       (X6) = REGISTER TYPE (BAX)                    DBG=IDP    2367    A
DBG=IDP            SX4    B0           (X4) = 0TR, WHERE T=0 (B REG)                 DBG=IDP    2368    A
DBG=IDP            SX1    X6-1RB                                                     DBG=IDP    2369    A
DBG=IDP            SX2    X6-1RA                                                     DBG=IDP    2370    A
DBG=IDP            SX3    X6-1RX                                                     DBG=IDP    2371    A
DBG=IDP            ZR     X1,ST=REG4   IF B REGISTER                                DBG=IDP    2372    A
DBG=IDP            SX4    010B         (X4) = 0TR, WHERE T=1 (A REG)                 DBG=IDP    2373    A
DBG=IDP            ZR     X2,ST=REG4   IF A REGISTER                                DBG=IDP    2374    A
DBG=IDP            SX4    020B         (X4) = 0TR, WHERE T=2 (X REG)                 DBG=IDP    2375    A
DBG=IDP            ZR     X3,ST=REG4   IF X REGISTER                                DBG=IDP    2376    A
DBG=IDP            EQ     IDP=ER                                                     DBG=IDP    2377    A
DBG=IDP                                                                              DBG=IDP    2378    A
DBG=IDP   ST=REG4  LX5    CHAR                                                       DBG=IDP    2379    A
DBG=IDP            MX0    -CHAR                                                      DBG=IDP    2380    A
DBG=IDP            BX6    -X0*X5                                                     DBG=IDP    2381    A
DBG=IDP            MX1    1R7-1R0+1                                                  DBG=IDP    2382    A
DBG=IDP            LX1    -1R0         (X1) = (0-7) SHIFT MASK                       DBG=IDP    2383    A
DBG=IDP            SB3    X6           (B3) = REGISTER NR (0-7)                      DBG=IDP    2384    A
DBG=IDP            LX7    X1,B3                                                      DBG=IDP    2385    A
DBG=IDP            LX5    5*CHAR                                                     DBG=IDP    2386    A
DBG=IDP            PL     X7,IDP=ER    IF NOT (0-7), ERROR...                        DBG=IDP    2387    A
DBG=IDP            MX0    -5*CHAR                                                    DBG=IDP    2388    A
DBG=IDP            BX7    -X0*X5                                                     DBG=IDP    2389    A
DBG=IDP            NZ     X7,IDP=ER    IF REST OF CHARS NON-ZERO, NOT A REGISTER     DBG=IDP    2390    A
DBG=IDP            SB6    B3-1R0       (B6) = REGISTER NR IN BINARY                  DBG=IDP    2391    A
DBG=IDP            MX0    60-3                                                       DBG=IDP    2392    A
DBG=IDP            BX7    X0*X4        STRIP OLD REGISTER NR                         DBG=IDP    2393    A
DBG=IDP            SX6    X7+B6        (X6) = NEW 0TR                                DBG=IDP    2394    A
DBG=IDP            SB4    X6                                                         DBG=IDP    2395    A
DBG=IDP            SA6    IDP0TR                                                     DBG=IDP    2396    A
DBG=IDP            SB7    020B                                                       DBG=IDP    2397    A
DBG=IDP            GE     B4,B7,ST=REG5     IF DUMPING X REGISTER                    DBG=IDP    2398    A
DBG=IDP            RJ     DAB          DUMP -A- OR -B- REGISTER                      DBG=IDP    2399    A
DBG=IDP            EQ     ST=REG2                                                    DBG=IDP    2400    A
DBG=IDP                                                                              DBG=IDP    2401    A
DBG=IDP   ST=REG5  RJ     DUX          DUMP -X- REGISTER                             DBG=IDP    2402    A
DBG=IDP            EQ     ST=REG2                                                    DBG=IDP    2403    A
DBG=IDP   ST=RES   SPACE  4,8                                                        DBG=IDP    2404    A
DBG=IDP  *         HERE TO PROCESS *RESET*.                                          DBG=IDP    2405    A
DBG=IDP                                                                              DBG=IDP    2406    A
DBG=IDP   ST=RES   SB6    IDPBA                                                      DBG=IDP    2407    A
DBG=IDP            RJ     CLZ          CLEAR *IDP* TABLE -- BREAK ADDRESSES          DBG=IDP    2408    A
DBG=IDP            SB6    IDPSET                                                     DBG=IDP    2409    A
DBG=IDP            RJ     CLZ          CLEAR *IDP* TABLE -- SET NAMES                DBG=IDP    2410    A
DBG=IDP            RJ     IST          (RE)INITIALIZE DEFAULT *SET* NAMES            DBG=IDP    2411    A
DBG=IDP            EQ     IDP=MN                                                     DBG=IDP    2412    A
DBG=IDP   ST=SET   SPACE  4,8                                                        DBG=IDP    2413    A
DBG=IDP  *         HERE TO PROCESS *SET NAME,ADDR*.                                  DBG=IDP    2414    A
DBG=IDP                                                                              DBG=IDP    2415    A
DBG=IDP   ST=SET   ZR     X5,ST=SET3   IF EOS ENCOUNTERED                           DBG=IDP    2416    A
DBG=IDP            SA5    A5+B1        (X5) = 42/0LNAME,18/0.VAR                     DBG=IDP    2417    A
DBG=IDP            SX6    X5-0.VAR                                                   DBG=IDP    2418    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            NZ     X6,IDP=ER   IF NOT A NAME TOKEN                          DBG=IDP    2419    A
DBG=IDP            MX0    7*CHAR                                                   DBG=IDP    2420    A
DBG=IDP            BX7    X0*X5                                                    DBG=IDP    2421    A
DBG=IDP            SA7    IDPNAM      SAVE 0LNAME                                  DBG=IDP    2422    A
DBG=IDP            SA5    A5+B1                                                    DBG=IDP    2423    A
DBG=IDP            RJ     PAS         PARSE SUBEXPRESSION -- ADDR                  DBG=IDP    2424    A
DBG=IDP            MX0    7*CHAR                                                   DBG=IDP    2425    A
DBG=IDP            SB6    IDPSET      (B6) = FWA OF *SET* TABLE                    DBG=IDP    2426    A
DBG=IDP            BX4    -X0*X1      (X4) = 42/0,18/VALUE OF *NAME*               DBG=IDP    2427    A
DBG=IDP            SA1    IDPNAM                                                   DBG=IDP    2428    A
DBG=IDP            RJ     SKT         SEARCH *SET* TABLE FOR NAME                  DBG=IDP    2429    A
DBG=IDP            IX6    X1+X4       (X6) = 42/0LNAME, 18/VALUE OF NAME           DBG=IDP    2430    A
DBG=IDP            ZR     X2,ST=SET2  IF NO FIND                                   DBG=IDP    2431    A
DBG=IDP            SA6    A2                                                       DBG=IDP    2432    A
DBG=IDP            EQ     IDP=MN                                                   DBG=IDP    2433    A
DBG=IDP                                                                            DBG=IDP    2434    A
DBG=IDP   ST=SET2  RJ     ADZ         ADD A WORD TO IDP TABLE                      DBG=IDP    2435    A
DBG=IDP            EQ     IDP=MN                                                   DBG=IDP    2436    A
DBG=IDP                                                                            DBG=IDP    2437    A
DBG=IDP   *        HERE TO LIST *SET* TABLE.                                       DBG=IDP    2438    A
DBG=IDP                                                                            DBG=IDP    2439    A
DBG=IDP   ST=SET3  SA5    IDPSET                                                   DBG=IDP    2440    A
DBG=IDP            RJ     LST         LIST SET TABLE                               DBG=IDP    2441    A
DBG=IDP            EQ     IDP=MN                                                   DBG=IDP    2442    A
DBG=IDP   ST=SNP   SPACE  4,8                                                      DBG=IDP    2443    A
DBG=IDP   *        HERE TO PROCESS *SNAP FWA,LWA,LEN*.                             DBG=IDP    2444    A
DBG=IDP                                                                            DBG=IDP    2445    A
DBG=IDP   ST=SNP   BSS    0                                                        DBG=IDP    2446    A
DBG=IDP            MX6    -1          (X6) = DEFAULT FWA (ERROR)          000480   DBG=IDP    2447    A
DBG=IDP            SX7    B1          (X7) = DEFAULT LEN                  000490   DBG=IDP    2448    A
DBG=IDP            SA6    PAT#FWA                                         000500   DBG=IDP    2449    A
DBG=IDP            SA7    PAT#LEN                                         000510   DBG=IDP    2450    A
DBG=IDP            RJ     PAT         PARSE *FWA,LWA,LEN* TRIPLE          000520   DBG=IDP    2451    A
DBG=IDP            MI     B2,IDP=ER   IF FWA, LWA, OR LEN IS BAD          000530   DBG=IDP    2452    A
DBG=IDP            SA2    AP=FWA      (X2) = FWA                          000540   DBG=IDP    2453    A
DBG=IDP            SA3    A2+B1       (X3) = LWA                          000550   DBG=IDP    2454    A
DBG=IDP            BX4    X4-X4       (X4) = LEN  (I.E. NO LEN)           000560   DBG=IDP    2455    A
DBG=IDP            RJ     DCM         DUMP CENTRAL MEMORY                          DBG=IDP    2456    A
DBG=IDP            PL     B2,IDP=MN   IF FWA,LWA,AND LEN OK                        DBG=IDP    2457    A
DBG=IDP            EQ     IDP=ER                                                   DBG=IDP    2458    A
DBG=IDP   ST=STO   SPACE  4,8                                                      DBG=IDP    2459    A
DBG=IDP   *        HERE TO PROCESS *STORE ADDR,C1,C2,C3,C4*.                       DBG=IDP    2460    A
DBG=IDP                                                                            DBG=IDP    2461    A
DBG=IDP   ST=STO   BSS    0                                                        DBG=IDP    2462    A
DBG=IDP            ZR     X5,IDP=ER   IF EOS ENCOUNTERED                           DBG=IDP    2463    A
DBG=IDP            RJ     PAS         PARSE SUBEXPRESSION -- ADDR TO STORE         DBG=IDP    2464    A
DBG=IDP            ZR     X3,IDP=ER   IF 1ST EXPRESSION *ADDR* IS NULL             DBG=IDP    2465    A
DBG=IDP            SX6    X1                                                       DBG=IDP    2466    A
DBG=IDP            SB2    X1                                                       DBG=IDP    2467    A
DBG=IDP            SA6    IDPSTA      SAVE STORE ADDR                              DBG=IDP    2468    A
DBG=IDP            RJ     CHK         CHECK CM ADDRESS -- STORE ADDR               DBG=IDP    2469    A
DBG=IDP            MI     B2,IDP=ER   IF ADDR IS BAD                               DBG=IDP    2470    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            MX6    0                                                    DBG=IDP    2471    A
DBG=IDP            SA6    A6+B1       CLEAR (IDPSTC) = NEW VALUE TO STORE      DBG=IDP    2472    A
DBG=IDP            SA6    A6+B1       CLEAR (IDPSTM) = NULL PARAMETER MASK     DBG=IDP    2473    A
DBG=IDP            ZR     X5,ST=STO4  IF STORE ADDR FOLLOWED BY EOS            DBG=IDP    2474    A
DBG=IDP                                                                        DBG=IDP    2475    A
DBG=IDP   ST=STO2  RJ     PAS         PARSE SUBEXPRESSION -- C-N-              DBG=IDP    2476    A
DBG=IDP            SA2    IDPSTC      (X2) = CURRENT VALUE TO BE STORED        DBG=IDP    2477    A
DBG=IDP            SA4    A2+B1       (X4) = CURRENT NULL PARAMETER MASK       DBG=IDP    2478    A
DBG=IDP            MX0    -15                                                  DBG=IDP    2479    A
DBG=IDP            LX2    15                                                   DBG=IDP    2480    A
DBG=IDP            LX4    15                                                   DBG=IDP    2481    A
DBG=IDP            BX6    X2+X1                                                DBG=IDP    2482    A
DBG=IDP            LX7    X4                                                   DBG=IDP    2483    A
DBG=IDP            SA6    A2                                                   DBG=IDP    2484    A
DBG=IDP            NZ     X3,ST=STO3  IF C-N- NOT NULL                         DBG=IDP    2485    A
DBG=IDP            BX7    -X0+X4                                               DBG=IDP    2486    A
DBG=IDP                                                                        DBG=IDP    2487    A
DBG=IDP   ST=STO3  SA7    A4                                                   DBG=IDP    2488    A
DBG=IDP            NZ     X5,ST=STO2  IF NOT FINISHED ASSEMBLING STORE VALUE   DBG=IDP    2489    A
DBG=IDP                                                                        DBG=IDP    2490    A
DBG=IDP   *        HERE IF FINISHED ASSEMBLING STORE VALUE.                    DBG=IDP    2491    A
DBG=IDP                                                                        DBG=IDP    2492    A
DBG=IDP   ST=STO4  SA1    IDPSTA      (X1) = STORE ADDR                        DBG=IDP    2493    A
DBG=IDP            SA2    A1+B1       (X2) = NEW ASSEMBLED STORE VALUE         DBG=IDP    2494    A
DBG=IDP            SA3    A2+B1       (X3) = NULL PARAMETER MASK               DBG=IDP    2495    A
DBG=IDP            SA1    X1          (X1) = OLD CONTENTS OF STORE ADDR        DBG=IDP    2496    A
DBG=IDP            BX6    X3*X1                                                DBG=IDP    2497    A
DBG=IDP            BX7    X2+X6       (X7) = NEW CONTENTS OF STORE ADDR        DBG=IDP    2498    A
DBG=IDP            SA7    A1          NEW REPLACES OLD                         DBG=IDP    2499    A
DBG=IDP            SA7    A2                                                   DBG=IDP    2500    A
DBG=IDP            RJ     =XWOD       CONVERT A WORD OF BINARY TO OCTAL DPC--OLD DBG=IDP  2501    A
DBG=IDP            SA6    IDPMSG4+1                                            DBG=IDP    2502    A
DBG=IDP            SA7    A6+B1                                                DBG=IDP    2503    A
DBG=IDP            SA1    IDPSTC      (X1) = NEW CONTENTS OF STORE ADDR        DBG=IDP    2504    A
DBG=IDP            RJ     =XWOD       CONVERT A WORD OF BINARY TO OCTAL DPC--NEW DBG=IDP  2505    A
DBG=IDP            SA6    IDPMSG5+1                                            DBG=IDP    2506    A
DBG=IDP            SA7    A6+B1                                                DBG=IDP    2507    A
DBG=IDP            PRIDP  IDPMSG4,4                                            DBG=IDP    2508    A
DBG=IDP            PRIDP  IDPMSG5,4                                            DBG=IDP    2509    A
DBG=IDP            EQ     IDP=MN                                               DBG=IDP    2510    A
DBG=IDP   ST=STP   SPACE  4,8                                                  DBG=IDP    2511    A
DBG=IDP   *        HERE TO PROCESS *STEP L,RJ*,                                DBG=IDP    2512    A
DBG=IDP   *                    OR *STPL ADDR1,L,RJ* FORM,                      DBG=IDP    2513    A
DBG=IDP   *                    OR *STEQ ADDR1,ADDR2,L,RJ* FORM.                DBG=IDP    2514    A
DBG=IDP                                                                        DBG=IDP    2515    A
DBG=IDP   ST=STP   SX6    B7+IDPKEY-IDPKST  (X6) = STEP TYPE CODE (XTC)        DBG=IDP    2516    A
DBG=IDP            SA6    IDPXTC                                               DBG=IDP    2517    A
DBG=IDP            ZR     X6,ST=STP5  IF *STEP L,RJ* FORM                      DBG=IDP    2518    A
DBG=IDP            ZR     X5,ST=STP1  IF EOS ENCOUNTERED                       DBG=IDP    2519    A
DBG=IDP            RJ     PAS         PARSE SUBEXPRESSION-- ADDR1              DBG=IDP    2520    A
DBG=IDP            NZ     X3,ST=STP2  IF ADDR1 NOT NULL                        DBG=IDP    2521    A
DBG=IDP                                                                        DBG=IDP    2522    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP   ST=STP1  SX1    0           (X1) = DEFAULT ADDR1                      DBG=IDP    2523    A
DBG=IDP                                                                          DBG=IDP    2524    A
DBG=IDP   ST=STP2  SB2    X1                                                     DBG=IDP    2525    A
DBG=IDP            RJ     CHK         CHECK CM ADDRESS                          DBG=IDP    2526    A
DBG=IDP            MI     B2,IDP=ER   IF ADDR1 IS BAD                           DBG=IDP    2527    A
DBG=IDP            SX6    B2                                                     DBG=IDP    2528    A
DBG=IDP            MX3    0                                                      DBG=IDP    2529    A
DBG=IDP            SA6    IDPXA1                                                 DBG=IDP    2530    A
DBG=IDP            SA2    IDPXTC                                                 DBG=IDP    2531    A
DBG=IDP            SX7    X2-XTC=EQ                                              DBG=IDP    2532    A
DBG=IDP            MI     X7,ST=STP3  IF THIS IS *STPL ADDR1,L,RJ* FORM         DBG=IDP    2533    A
DBG=IDP                                                                          DBG=IDP    2534    A
DBG=IDP   *        ASSEMBLE ADDR2.                                               DBG=IDP    2535    A
DBG=IDP                                                                          DBG=IDP    2536    A
DBG=IDP            ZR     X5,ST=STP3  IF ADDR1 FOLLOWED BY EOS                  DBG=IDP    2537    A
DBG=IDP            RJ     PAS         PARSE SUBEXPRESSION-- ADDR2               DBG=IDP    2538    A
DBG=IDP                                                                          DBG=IDP    2539    A
DBG=IDP   ST=STP3  NZ     X3,ST=STP4  IF ADDR2 NOT NULL                         DBG=IDP    2540    A
DBG=IDP            SA1    IDPXA1      (X1) = DEFAULT ADDR2                      DBG=IDP    2541    A
DBG=IDP                                                                          DBG=IDP    2542    A
DBG=IDP   ST=STP4  SB2    X1                                                     DBG=IDP    2543    A
DBG=IDP            SX6    B2                                                     DBG=IDP    2544    A
DBG=IDP            RJ     CHK         CHECK CM ADDRESS                          DBG=IDP    2545    A
DBG=IDP            MI     B2,IDP=ER   IF ADDR2 IS BAD                           DBG=IDP    2546    A
DBG=IDP            SA6    IDPXA2                                                 DBG=IDP    2547    A
DBG=IDP                                                                          DBG=IDP    2548    A
DBG=IDP   *        HERE TO PROCESS *L,RJ* OPTIONS LIST.                          DBG=IDP    2549    A
DBG=IDP                                                                          DBG=IDP    2550    A
DBG=IDP   ST=STP5  ZR     X5,ST=STP6  IF EOS ENCOUNTERED                        DBG=IDP    2551    A
DBG=IDP            SB6    IDPKSTO     (B6) = FWA OF STEP OPTIONS KEYWORD TABLE  DBG=IDP    2552    A
DBG=IDP            RJ     POL         PROCESS OPTIONS LIST                      DBG=IDP    2553    A
DBG=IDP            ZR     X3,ST=STP5  IF NULL PARAMETER                         DBG=IDP    2554    A
DBG=IDP            SA7    IDPXOP+B7                                              DBG=IDP    2555    A
DBG=IDP            EQ     ST=STP5                                                DBG=IDP    2556    A
DBG=IDP                                                                          DBG=IDP    2557    A
DBG=IDP   *        HERE TO SET UP FOR ACTUAL STEP.                               DBG=IDP    2558    A
DBG=IDP                                                                          DBG=IDP    2559    A
DBG=IDP   ST=STP6  SA1    IDPXTC                                                 DBG=IDP    2560    A
DBG=IDP            SA2    IDPXA2                                                 DBG=IDP    2561    A
DBG=IDP            MX6    0                                                      DBG=IDP    2562    A
DBG=IDP            SX3    X1-XTC=NR                                              DBG=IDP    2563    A
DBG=IDP            BX7    X2                                                     DBG=IDP    2564    A
DBG=IDP            SA6    IDPXRJ      SET TO *AT LEVEL 0 NOW*                   DBG=IDP    2565    A
DBG=IDP            NZ     X3,ST=STP7  IF NOT *STNR NR,L,RJ*                     DBG=IDP    2566    A
DBG=IDP            SA7    IDPXA1      (RE)SET STEP COUNT                        DBG=IDP    2567    A
DBG=IDP                                                                          DBG=IDP    2568    A
DBG=IDP   ST=STP7  SA1    IDPXOL      (X1) = /L/ OPTION FLAG                    DBG=IDP    2569    A
DBG=IDP            SA2    A1+B1       (X2) = /RJ/ OPTION FLAG                   DBG=IDP    2570    A
DBG=IDP            SX6    B1          SET TO *LIST ON*                          DBG=IDP    2571    A
DBG=IDP            SA3    IDPXRJ                                                 DBG=IDP    2572    A
DBG=IDP            NZ     X2,ST=STP7A IF /RJ/ TRACING ON                        DBG=IDP    2573    A
DBG=IDP            ZR     X3,ST=STP7A IF AT LEVEL 0 (I.E. NO ACTIVE *RJ*)       DBG=IDP    2574    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            SX6    B0          SET TO *LIST OFF*                          DBG=IDP    2575    A
DBG=IDP                                                                           DBG=IDP    2576    A
DBG=IDP   ST=STP7A BX7    X6*X1                                                   DBG=IDP    2577    A
DBG=IDP            SA7    IDPXLST                                                 DBG=IDP    2578    A
DBG=IDP                                                                           DBG=IDP    2579    A
DBG=IDP   *        HERE TO CHECK STEP CONDITIONS.                                 DBG=IDP    2580    A
DBG=IDP                                                                           DBG=IDP    2581    A
DBG=IDP   ST=STP8  SA3    IDPXTC      (X3) = STEP TYPE CODE                       DBG=IDP    2582    A
DBG=IDP            SA1    IDPXA1      (X1) = A1                                   DBG=IDP    2583    A
DBG=IDP            SA2    A1+B1       (X2) = A2                                   DBG=IDP    2584    A
DBG=IDP            SB5    X3                                                      DBG=IDP    2585    A
DBG=IDP            SX6    X3-XTC=NR                                               DBG=IDP    2586    A
DBG=IDP            SX0    X3-XTC=AR                                               DBG=IDP    2587    A
DBG=IDP            SX7    X3-XTC=RNG                                              DBG=IDP    2588    A
DBG=IDP            ZR     X3,ST=STP9  IF *STEP L,RJ*                              DBG=IDP    2589    A
DBG=IDP            ZR     X6,ST=STP10 IF *STNR NR,L,RJ*                           DBG=IDP    2590    A
DBG=IDP            ZR     X0,ST=STP9  IF *STAR ADDR1,ADDR2,L,RJ*                  DBG=IDP    2591    A
DBG=IDP            ZR     X7,ST=STP12 IF *STRANGE ADDR1,ADDR2,L,RJ*               DBG=IDP    2592    A
DBG=IDP                                                                           DBG=IDP    2593    A
DBG=IDP            SA1    X1          (X1) = C1                                   DBG=IDP    2594    A
DBG=IDP            SA2    X2          (X2) = C2                                   DBG=IDP    2595    A
DBG=IDP            RJ     CBC         CHECK STEP CONDITION                        DBG=IDP    2596    A
DBG=IDP            ZR     B6,IDP=MN   IF CONDITION NOT SATISFIED                  DBG=IDP    2597    A
DBG=IDP                                                                           DBG=IDP    2598    A
DBG=IDP   ST=STP9  RJ     STP         STEP ONE INSTRUCTION                        DBG=IDP    2599    A
DBG=IDP            SA1    IDPXTC                                                  DBG=IDP    2600    A
DBG=IDP            SA2    IDPXRJ                                                  DBG=IDP    2601    A
DBG=IDP            SA3    IDPXORJ                                                 DBG=IDP    2602    A
DBG=IDP            SX6    X1-XTC=AR                                               DBG=IDP    2603    A
DBG=IDP            ZR     X6,ST=STP9A IF *STAR ADDR1,ADDR2,L,RJ*                  DBG=IDP    2604    A
DBG=IDP            NZ     X1,ST=STP7  IF NOT *STEP L,RJ*                          DBG=IDP    2605    A
DBG=IDP                                                                           DBG=IDP    2606    A
DBG=IDP   *        HERE IF *STEP L,RJ*.                                           DBG=IDP    2607    A
DBG=IDP                                                                           DBG=IDP    2608    A
DBG=IDP            NZ     X3,IDP=MN   IF /RJ/ SELECTED                            DBG=IDP    2609    A
DBG=IDP            ZR     X2,IDP=MN   IF AT LEVEL 0                               DBG=IDP    2610    A
DBG=IDP            EQ     ST=STP7                                                 DBG=IDP    2611    A
DBG=IDP                                                                           DBG=IDP    2612    A
DBG=IDP   *        HERE TO PROCESS *STAR ADDR1,ADDR2,L,RJ*.                       DBG=IDP    2613    A
DBG=IDP                                                                           DBG=IDP    2614    A
DBG=IDP   ST=STP9A SA1    IDPXAR                                                  DBG=IDP    2615    A
DBG=IDP            SA2    IDPXA1                                                  DBG=IDP    2616    A
DBG=IDP            SA3    IDPXA2                                                  DBG=IDP    2617    A
DBG=IDP            SA4    IDPPREG                                                 DBG=IDP    2618    A
DBG=IDP            IX6    X1-X2                                                   DBG=IDP    2619    A
DBG=IDP            IX7    X3-X1                                                   DBG=IDP    2620    A
DBG=IDP            MI     X6,ST=STP7  IF ADDR BELOW LOWER LIMIT                   DBG=IDP    2621    A
DBG=IDP            MI     X7,ST=STP7  IF ADDR ABOVE UPPER LIMIT                   DBG=IDP    2622    A
DBG=IDP            EQ     IDP=MN                                                  DBG=IDP    2623    A
DBG=IDP                                                                           DBG=IDP    2624    A
DBG=IDP   *        HERE TO PROCESS *STNR NR,L,RJ*.                                DBG=IDP    2625    A
DBG=IDP                                                                           DBG=IDP    2626    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP   ST=STP10 SA2    IDPXORJ                                          DBG=IDP     2627     A
DBG=IDP            SA3    IDPXRJ                                           DBG=IDP     2628     A
DBG=IDP            SX6    X1-1                                             DBG=IDP     2629     A
DBG=IDP            NZ     X2,ST=STP11 IF /RJ/ SELECTED                     DBG=IDP     2630     A
DBG=IDP            NZ     X3,ST=STP9  IF NOT AT LEVEL 0                    DBG=IDP     2631     A
DBG=IDP                                                                    DBG=IDP     2632     A
DBG=IDP   ST=STP11 MI     X6,IDP=MN   IF NO MORE INSTRUCTIONS TO STEP      DBG=IDP     2633     A
DBG=IDP            SA6    A1                                               DBG=IDP     2634     A
DBG=IDP            EQ     ST=STP9                                          DBG=IDP     2635     A
DBG=IDP                                                                    DBG=IDP     2636     A
DBG=IDP  *         HERE TO PROCESS *STRANGE ADDR1,ADDR2,L,RJ*.              DBG=IDP     2637     A
DBG=IDP                                                                    DBG=IDP     2638     A
DBG=IDP   ST=STP12 SA4    IDPPREG                                          DBG=IDP     2639     A
DBG=IDP            IX6    X4-X1                                            DBG=IDP     2640     A
DBG=IDP            IX7    X2-X4                                            DBG=IDP     2641     A
DBG=IDP            MI     X6,IDP=MN   IF P BELOW LOWER RANGE               DBG=IDP     2642     A
DBG=IDP            MI     X7,IDP=MN   IF P ABOVE UPPER RANGE               DBG=IDP     2643     A
DBG=IDP            EQ     ST=STP9                                          DBG=IDP     2644     A
DBG=IDP   ST=UBK   SPACE  4,8                                              DBG=IDP     2645     A
DBG=IDP  *         HERE TO PROCESS *UNBREAK ADDR1,...,ADDRN*.               DBG=IDP     2646     A
DBG=IDP                                                                    DBG=IDP     2647     A
DBG=IDP   ST=UBK   BSS    0                                                DBG=IDP     2648     A
DBG=IDP            ZR     X5,IDP=MN   IF EOS ENCOUNTERED                   DBG=IDP     2649     A
DBG=IDP            RJ     PAS         PARSE SUBEXPRESSION-- ADDR           DBG=IDP     2650     A
DBG=IDP            ZR     X3,ST=UBK   IF ADDR IS NULL                      DBG=IDP     2651     A
DBG=IDP            SB6    IDPBA                                            DBG=IDP     2652     A
DBG=IDP            MX0    -18                                              DBG=IDP     2653     A
DBG=IDP            BX0    -X0                                              DBG=IDP     2654     A
DBG=IDP            RJ     SKT         SEARCH FOR *ADDR*                    DBG=IDP     2655     A
DBG=IDP            ZR     X2,IDP=ER   IF NO FIND                           DBG=IDP     2656     A
DBG=IDP            SB6    B7+B7                                            DBG=IDP     2657     A
DBG=IDP            SA4    IDPBC+B6                                         DBG=IDP     2658     A
DBG=IDP            MX6    -1                                               DBG=IDP     2659     A
DBG=IDP            BX7    X4                                               DBG=IDP     2660     A
DBG=IDP            SA6    A2          SET TO *THIS ENTRY IS AVAILABLE*     DBG=IDP     2661     A
DBG=IDP            SA7    X2          RESTORE PREVIOUS CONTENTS OF BREAK ADDR DBG=IDP  2662     A
DBG=IDP            EQ     ST=UBK      CONTINUE...                          DBG=IDP     2663     A
DBG=IDP   ST=UST   SPACE  4,8                                              DBG=IDP     2664     A
DBG=IDP  *         HERE TO PROCESS *UNSET NAME1,...,NAMEN*.                 DBG=IDP     2665     A
DBG=IDP                                                                    DBG=IDP     2666     A
DBG=IDP   ST=UST   BSS    0                                                DBG=IDP     2667     A
DBG=IDP                                                                    DBG=IDP     2668     A
DBG=IDP   ST=UST2  ZR     X5,IDP=MN   IF EOS ENCOUNTERED                   DBG=IDP     2669     A
DBG=IDP            SA5    A5+B1                                            DBG=IDP     2670     A
DBG=IDP            NO                                                      DBG=IDP     2671     A
DBG=IDP            SX6    X5-O.VAR                                         DBG=IDP     2672     A
DBG=IDP            NZ     X6,ST=UST2  IF NOT A NAME TOKEN                  DBG=IDP     2673     A
DBG=IDP            SB6    IDPSET                                           DBG=IDP     2674     A
DBG=IDP            MX0    7*CHAR                                           DBG=IDP     2675     A
DBG=IDP            BX1    X5                                               DBG=IDP     2676     A
DBG=IDP            RJ     SKT         SEARCH FOR NAME                      DBG=IDP     2677     A
DBG=IDP            ZR     X2,ST=UST2  IF NO FIND                           DBG=IDP     2678     A
```

```
         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            SX6    -B1                                                  DBG=IDP    2679    A
DBG=IDP            SA6    A2          SET TO *THIS ENTRY IS AVAILABLE*         DBG=IDP    2680    A
DBG=IDP            EQ     ST=UST2                                              DBG=IDP    2681    A
DBG=IDP   ST=WHR   SPACE  4,8                                                  DBG=IDP    2682    A
DBG=IDP  *         HERE TO PROCESS *WHERE ADDR1,...,ADDRN*.                    DBG=IDP    2683    A
DBG=IDP                                                                        DBG=IDP    2684    A
DBG=IDP   ST=WHR   BSS    0                                                    DBG=IDP    2685    A
DBG=IDP            ZR     X5,ST=WHR2A IF EOS ENCOUNTERED                       DBG=IDP    2686    A
DBG=IDP                                                                        DBG=IDP    2687    A
DBG=IDP   ST=WHR2  RJ     PAS         PARSE SUBEXPRESSION                      DBG=IDP    2688    A
DBG=IDP            NZ     X3,ST=WHR3  IF ADDR NOT NULL                         DBG=IDP    2689    A
DBG=IDP                                                                        DBG=IDP    2690    A
DBG=IDP   ST=WHR2A SA1    IDPPREG                                              DBG=IDP    2691    A
DBG=IDP                                                                        DBG=IDP    2692    A
DBG=IDP   ST=WHR3  MX0    -18                                                  DBG=IDP    2693    A
DBG=IDP            BX1    -X0*X1                                               DBG=IDP    2694    A
DBG=IDP            LX0    X1          SAVE (X1) = ADDR                         DBG=IDP    2695    A
DBG=IDP            RJ     COD         CONVERT BINARY TO OCTAL DPC              DBG=IDP    2696    A
DBG=IDP            LX6    9*CHAR-6*CHAR                                        DBG=IDP    2697    A
DBG=IDP            BX1    X0                                                   DBG=IDP    2698    A
DBG=IDP            SA6    SNAPLNE                                              DBG=IDP    2699    A
DBG=IDP                                                                        DBG=IDP    2700    A
DBG=IDP   #FRA     IF     DEF,FRA=                                             DBG=IDP    2701    A
DBG=IDP            SB7    ST=WHR4     (B7) = RETURN ADDRESS FOR *FRA=*         DBG=IDP    2702    A
DBG=IDP            EQ     =XFRA=      FIND RELATIVE ADDRESS                    DBG=IDP    2703    A
DBG=IDP                                                                        DBG=IDP    2704    A
DBG=IDP   ST=WHR4  SA6    A6+B1                                                DBG=IDP    2705    A
DBG=IDP            SA7    A6+B1                                                DBG=IDP    2706    A
DBG=IDP                                                                        DBG=IDP    2707    A
DBG=IDP   #FRA     ELSE                                                         DBG=IDP    2708    A
DBG=IDP            MX6    0                                                    DBG=IDP    2709    A
DBG=IDP            SA6    A6+B1       MARK EOL                                 DBG=IDP    2710    A
DBG=IDP   #FRA     ENDIF                                                        DBG=IDP    2711    A
DBG=IDP                                                                        DBG=IDP    2712    A
DBG=IDP            PRIDP  SNAPLNE                                              DBG=IDP    2713    A
DBG=IDP            NZ     X5,ST=WHR2  IF NOT EOS TOKEN                         DBG=IDP    2714    A
DBG=IDP            EQ     IDP=MN                                               DBG=IDP    2715    A
DBG=IDP   ST=XEQ   SPACE  4,8                                                  DBG=IDP    2716    A
DBG=IDP  *         HERE TO PROCESS *XEQ*.                                      DBG=IDP    2717    A
DBG=IDP                                                                        DBG=IDP    2718    A
DBG=IDP   ST=XEQ   =      ST=STP6                                              DBG=IDP    2719    A
DBG=IDP   ST=XFR   SPACE  4,8                                                  DBG=IDP    2720    A
DBG=IDP  *         HERE TO PROCESS *XFER NR*.                                  DBG=IDP    2721    A
DBG=IDP                                                                        DBG=IDP    2722    A
DBG=IDP   ST=XFR   ZR     X5,ST=XFR1  IF EOS ENCOUNTERED                       DBG=IDP    2723    A
DBG=IDP            RJ     PAS         PARSE SUBEXPRESSION -- NR                DBG=IDP    2724    A
DBG=IDP            NZ     X3,ST=XFR2  IF NR NOT NULL                           DBG=IDP    2725    A
DBG=IDP                                                                        DBG=IDP    2726    A
DBG=IDP   ST=XFR1  SX6    L.XFT       (X6) = DEFAULT NR = ALL ENTRIES          DBG=IDP    2727    A
DBG=IDP                                                                        DBG=IDP    2728    A
DBG=IDP   ST=XFR2  RJ     LXT         LIST XFER TABLE                          DBG=IDP    2729    A
DBG=IDP            EQ     IDP=MN                                               DBG=IDP    2730    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP   ST=XNR   SPACE  4,8                                                  DBG=IDP    2731    A
DBG=IDP  *         HERE TO PROCESS *STNR,NR,L,RJ*.                            DBG=IDP    2732    A
DBG=IDP                                                                       DBG=IDP    2733    A
DBG=IDP   ST=XNR   SX6    XTC=NR                                              DBG=IDP    2734    A
DBG=IDP            SA6    IDPXTC                                              DBG=IDP    2735    A
DBG=IDP            ZR     X5,ST=STP5  IF EOS ENCOUNTERED                      DBG=IDP    2736    A
DBG=IDP            RJ     PAS         PARSE SUBEXPRESSION-- NR                DBG=IDP    2737    A
DBG=IDP            ZR     X3,ST=STP5  IF NR IS NULL                           DBG=IDP    2738    A
DBG=IDP            SA6    IDPXA1                                              DBG=IDP    2739    A
DBG=IDP            SA6    IDPXA2                                              DBG=IDP    2740    A
DBG=IDP            EQ     ST=STP5     REJOIN NORMAL STEP PROCESSING FLOW...   DBG=IDP    2741    A
DBG=IDP   ADZ      SPACE  4,8                                                  DBG=IDP    2742    A
DBG=IDP  **        ADZ - ADD A WORD TO IDP TABLE.                              DBG=IDP    2743    A
DBG=IDP  *                                                                     DBG=IDP    2744    A
DBG=IDP  *                                                                     DBG=IDP    2745    A
DBG=IDP  *                THIS ROUTINE WILL ADD A SINGLE WORD TO AN IDP FIXED  DBG=IDP    2746    A
DBG=IDP  *         LENGTH TABLE THAT IS TERMINATED BY A ZERO WORD, AND WHERE   DBG=IDP    2747    A
DBG=IDP  *         AN AVAILABLE SPACE IS MARKED BY A -1.                       DBG=IDP    2748    A
DBG=IDP  *                                                                     DBG=IDP    2749    A
DBG=IDP  *         ENTRY  (X6)   =  WORD TO BE ADDED TO TABLE                  DBG=IDP    2750    A
DBG=IDP  *                (B6)   =  FWA OF TABLE                               DBG=IDP    2751    A
DBG=IDP  *                                                                     DBG=IDP    2752    A
DBG=IDP  *         EXIT   TO *IDP=OV* IF TABLE OVERFLOW                        DBG=IDP    2753    A
DBG=IDP  *                (X1)   =  -1 IF ENTRY ALREADY IN TABLE, ELSE         DBG=IDP    2754    A
DBG=IDP  *                       =  ENTRY THAT WAS MADE IN TABLE               DBG=IDP    2755    A
DBG=IDP  *                (B7)   =  ORDINAL OF ENTRY THAT WAS MADE OR OF MATCHING  DBG=IDP    2756    A
DBG=IDP  *                            ENTRY                                    DBG=IDP    2757    A
DBG=IDP  *                                                                     DBG=IDP    2758    A
DBG=IDP  *         USES   X - 0,1,2,3,6,7                                      DBG=IDP    2759    A
DBG=IDP  *                A - 1,2                                              DBG=IDP    2760    A
DBG=IDP  *                B - 7                                                DBG=IDP    2761    A
DBG=IDP  *                                                                     DBG=IDP    2762    A
DBG=IDP  *         CALLS  IDP.ER,PRIDP,SKT                                     DBG=IDP    2763    A
DBG=IDP                                                                        DBG=IDP    2764    A
DBG=IDP                                                                        DBG=IDP    2765    A
DBG=IDP   ADZ      SUBR                ** ENTRY/EXIT **                        DBG=IDP    2766    A
DBG=IDP            BX1    X6                                                   DBG=IDP    2767    A
DBG=IDP            MX0    60                                                   DBG=IDP    2768    A
DBG=IDP            SA6    ADZA        SAVE (X6)                                DBG=IDP    2769    A
DBG=IDP            RJ     SKT         SEARCH TO SEE IF ENTRY ALREADY IN TABLE  DBG=IDP    2770    A
DBG=IDP            SX1    -B1                                                  DBG=IDP    2771    A
DBG=IDP            NZ     X2,EXIT.    IF ENTRY ALREADY IN TABLE                DBG=IDP    2772    A
DBG=IDP            RJ     SKT         SEARCH FOR AVAILABLE SPACE               DBG=IDP    2773    A
DBG=IDP            ZR     X2,IDP=OV   IF NO AVAILABLE SPACE LEFT               DBG=IDP    2774    A
DBG=IDP            SA1    ADZA                                                 DBG=IDP    2775    A
DBG=IDP            BX6    X1                                                   DBG=IDP    2776    A
DBG=IDP            SA6    A2          MAKE NEW TABLE ENTRY                     DBG=IDP    2777    A
DBG=IDP            EQ     EXIT.                                                DBG=IDP    2778    A
DBG=IDP                                                                        DBG=IDP    2779    A
DBG=IDP                                                                        DBG=IDP    2780    A
DBG=IDP   ADZA     BSSZ   1                                                    DBG=IDP    2781    A
DBG=IDP   BRK      SPACE  4,8                                                  DBG=IDP    2782    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  **        BRK - BREAK PROCESSOR.                                    DBG=IDP    2783    A
DBG=IDP  *                                                                   DBG=IDP    2784    A
DBG=IDP  *                                                                   DBG=IDP    2785    A
DBG=IDP  *         1. IF THIS IS AN ASSEMBLED BREAK (NOT *IDP* GENERATED),   DBG=IDP    2786    A
DBG=IDP  *            THEN CONTROL IS MERELY RETURNED TO CALLER.             DBG=IDP    2787    A
DBG=IDP  *                                                                   DBG=IDP    2788    A
DBG=IDP  *         2. IF THIS IS AN *IDP* GENERATED BREAK, THE 2 PARALLEL TABLES,  DBG=IDP  2789  A
DBG=IDP  *            *IDPBA/IDPBC*, ARE QUERIED FOR INFORMATION THAT WILL BE  DBG=IDP  2790  A
DBG=IDP  *            PLUGGED INTO A DUMMY PARAMETER LIST FOR THIS BREAK. THE  DBG=IDP  2791  A
DBG=IDP  *            PREVIOUS CONTENTS OF THE BREAK ADDR ARE PICKED UP FROM   DBG=IDP  2792  A
DBG=IDP  *            *IDPBC* AND PLUGGED AT *ST=ENDX* SO THAT THEY CAN BE     DBG=IDP  2793  A
DBG=IDP  *            EXECUTED BEFORE EXITING *IDP*.                           DBG=IDP  2794  A
DBG=IDP  *                                                                   DBG=IDP    2795    A
DBG=IDP  *         TABLE FORMATS ARE--                                       DBG=IDP    2796    A
DBG=IDP  *                                                                   DBG=IDP    2797    A
DBG=IDP  *         IDPBA ENTRY AT (IDPBA+ORDINAL)                            DBG=IDP    2798    A
DBG=IDP  *                                                                   DBG=IDP    2799    A
DBG=IDP  *         VFD    6/BREAK TYPE CODE (BTC)                             DBG=IDP    2800    A
DBG=IDP  *         VFD    18/ ADDR1                                           DBG=IDP    2801    A
DBG=IDP  *         VFD    18/ ADDR2   FOR A CONDITIONAL BREAK (BRNE,...) THE  DBG=IDP  2802  A
DBG=IDP  *                              CONTENTS OF THE CORE LOCATION ADDR1 WILL  DBG=IDP  2803  A
DBG=IDP  *                              BE COMPARED TO THE CONTENTS OF CORE      DBG=IDP  2804  A
DBG=IDP  *                              LOCATION ADDR2.                          DBG=IDP  2805  A
DBG=IDP  *         VFD    18/ BREAK ADDR                                      DBG=IDP    2806    A
DBG=IDP  *                                                                   DBG=IDP    2807    A
DBG=IDP  *         IDPBC ENTRY AT (IDPBC+2*ORDINAL)                           DBG=IDP    2808    A
DBG=IDP  *                                                                   DBG=IDP    2809    A
DBG=IDP  *         VFD    60/ PREVIOUS CONTENTS OF BREAK ADDR                 DBG=IDP    2810    A
DBG=IDP  *         VFD    15/ LL                                              DBG=IDP    2811    A
DBG=IDP  *         VFD    15/ UL                                              DBG=IDP    2812    A
DBG=IDP  *         VFD    12/ INC                                             DBG=IDP    2813    A
DBG=IDP  *         VFD    18/ SNAP COUNT                                      DBG=IDP    2814    A
DBG=IDP  *                                                                   DBG=IDP    2815    A
DBG=IDP  *            IF THIS IS A CONDITIONAL *IDP* GENERATED BREAK         DBG=IDP    2816    A
DBG=IDP  *            (BRPL,BREQ,...), THE RELATIONAL CONDITION IS EVALUATED. DBG=IDP    2817    A
DBG=IDP  *            A TRUE RESULT MEANS THE BREAK WILL BE HONORED, A FALSE THAT  DBG=IDP  2818  A
DBG=IDP  *            IT WILL NOT.                                           DBG=IDP    2819    A
DBG=IDP  *                                                                   DBG=IDP    2820    A
DBG=IDP  *         ENTRY  (X1)    =   (IDPPREG)  PSEUDO P REGISTER           DBG=IDP    2821    A
DBG=IDP  *                           (IDPPREG)-1 = ADDR OF IDP CALLER (BREAK AD)  DBG=IDP  2822  A
DBG=IDP  *                (A5,    =   (IDPPREG)-1                            DBG=IDP    2823    A
DBG=IDP  *                    X5) =   FWA OF PARAMETER LIST                  DBG=IDP    2824    A
DBG=IDP  *                        =   .ZR. IF *IDP* GENERATED BREAK          DBG=IDP    2825    A
DBG=IDP  *                                                                   DBG=IDP    2826    A
DBG=IDP  *         EXIT   (X5)    =   FWA OF PARAMETER LIST FOR THIS BREAK   DBG=IDP    2827    A
DBG=IDP  *                              (OR DUMMY PARAMETER LIST FOR *IDP* BREAK)  DBG=IDP  2828  A
DBG=IDP  *                        =   .ZR. IF NO BREAK THIS TIME             DBG=IDP    2829    A
DBG=IDP  *                                                                   DBG=IDP    2830    A
DBG=IDP  *         USES   X - ALL                                            DBG=IDP    2831    A
DBG=IDP  *                A - 1,2,3,6,7                                      DBG=IDP    2832    A
DBG=IDP  *                B - 5,6,7                                          DBG=IDP    2833    A
DBG=IDP  *                                                                   DBG=IDP    2834    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *         CALLS  CBC,SKT                                          DBG=IDP    2835    A
DBG=IDP                                                                    DBG=IDP    2836    A
DBG=IDP                                                                    DBG=IDP    2837    A
DBG=IDP   BRK      SUBR                ** ENTRY/EXIT **                    DBG=IDP    2838    A
DBG=IDP            SX6    X5                                               DBG=IDP    2839    A
DBG=IDP            MX7    -0                                               DBG=IDP    2840    A
DBG=IDP            SX1    X1-1        (X1) = BREAK ADDR                    DBG=IDP    2841    A
DBG=IDP            SA6    FWAPARM                                          DBG=IDP    2842    A
DBG=IDP            NZ     X5,EXIT.    IF NOT *IDP* GENERATED BREAK         DBG=IDP    2843    A
DBG=IDP                                                                    DBG=IDP    2844    A
DBG=IDP  *         SET UP DUMMY APLIST FOR *IDP* GENERATED BREAK.          DBG=IDP    2845    A
DBG=IDP                                                                    DBG=IDP    2846    A
DBG=IDP            SX6    APL         (X6) = FWA OF DUMMY APLIST           DBG=IDP    2847    A
DBG=IDP            SA7    AP=HDR      SET TO *IDP GENERATED BREAK HEADER*  DBG=IDP    2848    A
DBG=IDP            SA6    A6                                               DBG=IDP    2849    A
DBG=IDP            MX0    -18                                              DBG=IDP    2850    A
DBG=IDP            SX5    X6                                               DBG=IDP    2851    A
DBG=IDP            BX0    -X0                                              DBG=IDP    2852    A
DBG=IDP            SB6    IDPBA                                            DBG=IDP    2853    A
DBG=IDP            RJ     SKT         SEARCH FOR BREAK ADDR                DBG=IDP    2854    A
DBG=IDP            ZR     X2,*+4S15   IF NO FIND, IRRETRIEVABLE ERROR...   DBG=IDP    2855    A
DBG=IDP            SB6    B7+B7                                            DBG=IDP    2856    A
DBG=IDP            SA3    IDPBC+B6+1  (X3) = 15/LL,15/UL,12/INC,18/CNT     DBG=IDP    2857    A
DBG=IDP            SX6    X3                                               DBG=IDP    2858    A
DBG=IDP            BX7    -X0*X3      (X7) = 15/LL,15/UL,12/INC,18/0       DBG=IDP    2859    A
DBG=IDP            SA6    AP=CNT                                           DBG=IDP    2860    A
DBG=IDP            SX4    X3+B1       (X4) = CNT+1                         DBG=IDP    2861    A
DBG=IDP            BX7    X7+X4       (X7) = 15/LL,15/UL,12/INC,18/CNT+1   DBG=IDP    2862    A
DBG=IDP            SA7    A3                                               DBG=IDP    2863    A
DBG=IDP            MX0    -15                                              DBG=IDP    2864    A
DBG=IDP            LX3    15                                               DBG=IDP    2865    A
DBG=IDP            BX6    -X0*X3      (X6) = LL                            DBG=IDP    2866    A
DBG=IDP            LX3    15                                               DBG=IDP    2867    A
DBG=IDP            BX7    -X0*X3      (X7) = UL                            DBG=IDP    2868    A
DBG=IDP            SA6    AP=LL                                            DBG=IDP    2869    A
DBG=IDP            SA7    A6+B1                                            DBG=IDP    2870    A
DBG=IDP            MX0    -12                                              DBG=IDP    2871    A
DBG=IDP            LX3    12                                               DBG=IDP    2872    A
DBG=IDP            BX6    -X0*X3      (X6) = INC                           DBG=IDP    2873    A
DBG=IDP            SA6    A7+B1                                            DBG=IDP    2874    A
DBG=IDP                                                                    DBG=IDP    2875    A
DBG=IDP  *         SET UP EXIT SO THAT PREVIOUS CONTENTS OF BREAK ADDR WILL BE  DBG=IDP  2876  A
DBG=IDP  *           EXECUTED BEFORE EXITING *IDP*.                        DBG=IDP    2877    A
DBG=IDP  *                                                                 DBG=IDP    2878    A
DBG=IDP            SA3    A3-B1       (X3) = PREVIOUS CONTENTS OF BREAK ADDR  DBG=IDP  2879  A
DBG=IDP            BX6    X3                                               DBG=IDP    2880    A
DBG=IDP            SA6    ST=ENDX                                          DBG=IDP    2881    A
DBG=IDP                                                                    DBG=IDP    2882    A
DBG=IDP  *         CHECK TO SEE IF BREAK CONDITION SATISFIED.              DBG=IDP    2883    A
DBG=IDP                                                                    DBG=IDP    2884    A
DBG=IDP            MX0    -6                                               DBG=IDP    2885    A
DBG=IDP            LX2    6                                                DBG=IDP    2886    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP          BX6    -X0*X2      (X6) = BREAK TYPE CODE (BTC)                DBG=IDP    2887    A
DBG=IDP          SB5    X6                                                      DBG=IDP    2888    A
DBG=IDP          MX0    -18                                                     DBG=IDP    2889    A
DBG=IDP          LX2    18                                                      DBG=IDP    2890    A
DBG=IDP          BX6    -X0*X2                                                  DBG=IDP    2891    A
DBG=IDP          SA1    X6          (X1) = (ADDR1)                              DBG=IDP    2892    A
DBG=IDP          LX2    18                                                      DBG=IDP    2893    A
DBG=IDP          BX6    -X0*X2                                                  DBG=IDP    2894    A
DBG=IDP          SA2    X6          (X2) = (ADDR2)                              DBG=IDP    2895    A
DBG=IDP          RJ     CBC         CHECK BREAK CONDITION                       DBG=IDP    2896    A
DBG=IDP          NZ     B6,EXIT.    IF CONDITION WAS SATISFIED (TRUE)           DBG=IDP    2897    A
DBG=IDP          MX5    0                                                       DBG=IDP    2898    A
DBG=IDP          EQ     EXIT.                                                   DBG=IDP    2899    A
DBG=IDP   BUB    SPACE  4,8                                                     DBG=IDP    2900    A
DBG=IDP  **      BUB - BURST/BUILD IDP COMMAND.                                 DBG=IDP    2901    A
DBG=IDP  *                                                                      DBG=IDP    2902    A
DBG=IDP  *                                                                      DBG=IDP    2903    A
DBG=IDP  *               THIS ROUTINE BURSTS AND BUILDS A PACKED (I.E. 10 CHARS DBG=IDP    2904    A
DBG=IDP  *       PER WORD) SOURCE LINE AT *IDPLIM* ET SEQ INTO ITS TOKEN FORM.  DBG=IDP    2905    A
DBG=IDP  *                                                                      DBG=IDP    2906    A
DBG=IDP  *       GENERAL TOKEN FORMAT IS --                                     DBG=IDP    2907    A
DBG=IDP  *                                                                      DBG=IDP    2908    A
DBG=IDP  *       VFD    42/0L_CHARS,18/TOKEN TYPE                               DBG=IDP    2909    A
DBG=IDP  *               CHARS = DISPLAY CODE CHARACTERS THAT CONSTITUTE THIS   DBG=IDP    2910    A
DBG=IDP  *                         TOKEN                                        DBG=IDP    2911    A
DBG=IDP  *               TOKEN TYPE = A TYPE CODE THAT IS USED TO DISTINGUISH   DBG=IDP    2912    A
DBG=IDP  *                            ONE TOKEN FROM ANOTHER. (NOTE--BIT 17     DBG=IDP    2913    A
DBG=IDP  *                            OF TOKEN TYPE MUST BE 0)                  DBG=IDP    2914    A
DBG=IDP  *                                                                      DBG=IDP    2915    A
DBG=IDP  *       TOKEN TYPES ARE --                                             DBG=IDP    2916    A
DBG=IDP  *                                                                      DBG=IDP    2917    A
DBG=IDP  *       O.VAR  VARIABLE OR NAME TOKEN  (ALPHANUMERIC)                  DBG=IDP    2918    A
DBG=IDP  *       O.CONS CONSTANT TOKEN  (NUMERIC)                               DBG=IDP    2919    A
DBG=IDP  *       O.SEP  SEPARATOR TOKEN (E.G. *,+-)                             DBG=IDP    2920    A
DBG=IDP  *                                                                      DBG=IDP    2921    A
DBG=IDP  *       E.G. IF ; IS THE END-OF-STMT CHARACTER --                      DBG=IDP    2922    A
DBG=IDP  *                                                                      DBG=IDP    2923    A
DBG=IDP  *       SNAP,NAME,,4;REG,X1                                            DBG=IDP    2924    A
DBG=IDP  *                                                                      DBG=IDP    2925    A
DBG=IDP  *       WOULD BECOME --                                                DBG=IDP    2926    A
DBG=IDP  *       VFD    42/OTHER ,18/LEN OF STMT = 7                            DBG=IDP    2927    A
DBG=IDP  *       VFD    42/4LSNAP,18/O.VAR                                      DBG=IDP    2928    A
DBG=IDP  *       VFD    42/1L,   ,18/O.SEP                                      DBG=IDP    2929    A
DBG=IDP  *       VFD    42/4LNAME,18/O.VAR                                      DBG=IDP    2930    A
DBG=IDP  *       VFD    42/1L,   ,18/O.SEP                                      DBG=IDP    2931    A
DBG=IDP  *       VFD    42/1L,   ,18/O.SEP                                      DBG=IDP    2932    A
DBG=IDP  *       VFD    42/1L4   ,18/O.CONS                                     DBG=IDP    2933    A
DBG=IDP  *       VFD    60/0  ** EOS **                                         DBG=IDP    2934    A
DBG=IDP  *                                                                      DBG=IDP    2935    A
DBG=IDP  *       VFD    42/OTHER ,18/LEN OF STMT = 4                            DBG=IDP    2936    A
DBG=IDP  *       VFD    42/3LREG ,18/O.VAR                                      DBG=IDP    2937    A
DBG=IDP  *       VFD    42/1L,   ,18/O.SEP                                      DBG=IDP    2938    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *         VFD    42/2LX1  ,18/0.VAR                                  DBG=IDP    2939    A
DBG=IDP  *         VFD    60/0  ** EOS **                                     DBG=IDP    2940    A
DBG=IDP  *                                                                    DBG=IDP    2941    A
DBG=IDP  *         ENTRY  (A5,X5)   = A+C OF 1ST WORD OF SOURCE LINE IMAGE    DBG=IDP    2942    A
DBG=IDP  *                              TO BE BURST/BUILT                     DBG=IDP    2943    A
DBG=IDP  *                (A6)+1    = ADDR TO STORE 1ST TOKEN THAT IS BURST/BUILT DBG=IDP  2944    A
DBG=IDP  *                (L=LIM)   = NR OF WORDS IN SOURCE LINE IMAGE AT     DBG=IDP    2945    A
DBG=IDP  *                              (IDPLIM) ET SEQ  (INCLUDES WORD       DBG=IDP    2946    A
DBG=IDP  *                              CONTAINING EOL MARK)                  DBG=IDP    2947    A
DBG=IDP  *                                                                    DBG=IDP    2948    A
DBG=IDP  *         EXIT   NONE                                                DBG=IDP    2949    A
DBG=IDP  *                                                                    DBG=IDP    2950    A
DBG=IDP  *         USES   ALL                                                 DBG=IDP    2951    A
DBG=IDP  *                                                                    DBG=IDP    2952    A
DBG=IDP  *         CALLS  NONE                                                DBG=IDP    2953    A
DBG=IDP                                                                       DBG=IDP    2954    A
DBG=IDP                                                                       DBG=IDP    2955    A
DBG=IDP   BUB      SUBR               ** ENTRY/EXIT **                        DBG=IDP    2956    A
DBG=IDP                                                                       DBG=IDP    2957    A
DBG=IDP  *         FIRST TIME ONLY INITIALIZATION.                            DBG=IDP    2958    A
DBG=IDP                                                                       DBG=IDP    2959    A
DBG=IDP            SX7    A6                                                  DBG=IDP    2960    A
DBG=IDP            MX0    -CHAR                                               DBG=IDP    2961    A
DBG=IDP            SA4    10*CHAR                                             DBG=IDP    2962    A
DBG=IDP            SA7    IDPSTMT                                             DBG=IDP    2963    A
DBG=IDP            SB3    CHAR                                                DBG=IDP    2964    A
DBG=IDP            SB6    A4                                                  DBG=IDP    2965    A
DBG=IDP            SA0    B0          SET TO *LENGTH=0 FIRST TIME THROUGH*    DBG=IDP    2966    A
DBG=IDP                                                                       DBG=IDP    2967    A
DBG=IDP  *         SET UP (B7) = NR OF WORDS TO BURST/BUILD (NOT INCLUDING WORD DBG=IDP  2968    A
DBG=IDP  *           CONTAINING EOL MARK).  I.E. (B7) WILL CONTAIN NR OF FULL DBG=IDP   2969    A
DBG=IDP  *           WORDS (10 CHARS EACH) TO BURST/BUILD.  THREE END-OF-LINE DBG=IDP   2970    A
DBG=IDP  *           CONDITIONS ARE SIGNIFICANT --                            DBG=IDP   2971    A
DBG=IDP  *                                                                    DBG=IDP    2972    A
DBG=IDP  *             (A5)+0         1     (X=ANY CHAR,0=00B)               DBG=IDP    2973    A
DBG=IDP  *                 XXXXXXXX00            (L=LIM) = 1, (B7) = 0       DBG=IDP    2974    A
DBG=IDP  *                +0         1                                       DBG=IDP    2975    A
DBG=IDP  *                 XXXXXXXXXX0000000000  (L=LIM) = 1, (B7) = 1       DBG=IDP    2976    A
DBG=IDP  *                +0         1                                       DBG=IDP    2977    A
DBG=IDP  *                 XXXXXXXXX00000000000  (L=LIM) = 2, (B7) = 0       DBG=IDP    2978    A
DBG=IDP                                                                       DBG=IDP    2979    A
DBG=IDP            SA1    =XL=LIM     (X1) = NR OF WORDS IN SOURCE LINE IMAGE DBG=IDP    2980    A
DBG=IDP  *                                (INCLUDES WORD CONTAINING EOL MRK)  DBG=IDP    2981    A
DBG=IDP            SB5    A5-2                                                DBG=IDP    2982    A
DBG=IDP            SB7    X1-1        (B7) = NR OF WORDS IN SOURCE LINE IMAGE DBG=IDP    2983    A
DBG=IDP  *                                (NOT INCLUDING WD CONTAINING EOL)   DBG=IDP    2984    A
DBG=IDP            SA2    B5+X1       (A2,X2) = A+C OF NEXT TO LAST WORD OF LINE DBG=IDP 2985    A
DBG=IDP            ZR     B7,BUB2     IF 11 CHAR (66 BIT) EOL NOT POSSIBLE    DBG=IDP    2986    A
DBG=IDP            BX6    -X0*X2                                              DBG=IDP    2987    A
DBG=IDP            NZ     X6,BUB2     IF NOT 11 CHAR (66 BIT) EOL MARK        DBG=IDP    2988    A
DBG=IDP            SB7    B7-B1                                               DBG=IDP    2989    A
DBG=IDP                                                                       DBG=IDP    2990    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *         INITIALIZE FOR BURST/BUILD OF KEYWORD.                          DBG=IDP    2991    A
DBG=IDP                                                                            DBG=IDP    2992    A
DBG=IDP   BUB2     SA2    MX=KEYW                                                  DBG=IDP    2993    A
DBG=IDP            MX1    0           SET TO *NOT SQUEEZING BLANKS*                DBG=IDP    2994    A
DBG=IDP            SB5    B0          SET TO *NOT SQUEEZING BLANKS*                DBG=IDP    2995    A
DBG=IDP            BX2    -X2                                                      DBG=IDP    2996    A
DBG=IDP            EQ     BUB2B                                                    DBG=IDP    2997    A
DBG=IDP                                                                            DBG=IDP    2998    A
DBG=IDP  *         INITIALIZE FOR BURST/BUILD OF NON-KEYWORD.                      DBG=IDP    2999    A
DBG=IDP                                                                            DBG=IDP    3000    A
DBG=IDP   BUB2A    SA1    =10H                                                     DBG=IDP    3001    A
DBG=IDP            SA2    MX=TOKN                                                  DBG=IDP    3002    A
DBG=IDP            SB5    1R +7777B-1RM                                            DBG=IDP    3003    A
DBG=IDP                                                                            DBG=IDP    3004    A
DBG=IDP  *         COMMON INITIALIZATION FOR BURST/BUILD.                          DBG=IDP    3005    A
DBG=IDP                                                                            DBG=IDP    3006    A
DBG=IDP   BUB2B    SA3    BUB=LEN                                                  DBG=IDP    3007    A
DBG=IDP            SX4    A0                                                       DBG=IDP    3008    A
DBG=IDP            IX7    X3-X4                                                    DBG=IDP    3009    A
DBG=IDP            SB4    8*CHAR-1*CHAR                                            DBG=IDP    3010    A
DBG=IDP            BX6    X6-X6                                                    DBG=IDP    3011    A
DBG=IDP            SA7    A3                                                       DBG=IDP    3012    A
DBG=IDP            NZ     B7,BUB4     IF 1ST WORD OF SOURCE LINE DOES NOT CONTAIN  DBG=IDP    3013    A
DBG=IDP  *                               EOL MARK, INTO THE FIRE...               DBG=IDP    3014    A
DBG=IDP                                                                            DBG=IDP    3015    A
DBG=IDP  *         HERE TO PROCESS WORD CONTAINING EOL MARK (I.E. LAST WORD OF     DBG=IDP    3016    A
DBG=IDP  *           SOURCE LINE) -- FIGURE OUT HOW MANY USEABLE (I.E. NON-EOL)    DBG=IDP    3017    A
DBG=IDP  *           BITS ARE IN FINAL WORD TO BE BURST/BUILT.                     DBG=IDP    3018    A
DBG=IDP                                                                            DBG=IDP    3019    A
DBG=IDP   BUB2C    MI     B7,BUB5     IF HAVE ALREADY PROCESSED LAST WORD OF LINE  DBG=IDP    3020    A
DBG=IDP            SA3    =40404040404040404040B                                   DBG=IDP    3021    A
DBG=IDP            SX4    B1                                                       DBG=IDP    3022    A
DBG=IDP            IX7    X5-X4       BORROW RIPPLES LEFT TO 1ST NON-ZERO BIT      DBG=IDP    3023    A
DBG=IDP            BX4    -X7+X5      (X4) = ALL TRAILING 0 BITS IN (X1) = 0 BITS  DBG=IDP    3024    A
DBG=IDP  *                                   ALL OTHER LEADING BITS = 1 BITS       DBG=IDP    3025    A
DBG=IDP            SB2    60-5        (B2) = RIGHT CIRCULAR SHIFT 5 BITS           DBG=IDP    3026    A
DBG=IDP            BX7    X3*X4       (X7) = 40B WHERE EACH CHAR IN LAST WORD IS   DBG=IDP    3027    A
DBG=IDP            LX4    X7,B2       (X4) = 01B WHERE EACH CHAR IN LAST WORD IS   DBG=IDP    3028    A
DBG=IDP            IX3    X7-X4       (X3) = 37B WHERE EACH CHAR IN LAST WORD IS   DBG=IDP    3029    A
DBG=IDP            BX4    X7+X3       (X4) = 77B WHERE EACH CHAR IN LAST WORD IS   DBG=IDP    3030    A
DBG=IDP            CX7    X4                                                       DBG=IDP    3031    A
DBG=IDP            SB7    B7-B1       (B7) = SET WORD COUNT TO -1 (I.E. BURSTING   DBG=IDP    3032    A
DBG=IDP  *                                   LAST WD) TO AVOID LOOPING AT BUB2B    DBG=IDP    3033    A
DBG=IDP            SB6    X7          (B6) = NR OF BITS TO BU/BU IN LAST WD (X5)   DBG=IDP    3034    A
DBG=IDP            EQ     BUB4        BURST/BUILD LAST WORD...                     DBG=IDP    3035    A
DBG=IDP                                                                            DBG=IDP    3036    A
DBG=IDP  *         BURST/BUILD LOOP  ** INSTACK **                                 DBG=IDP    3037    A
DBG=IDP                                                                            DBG=IDP    3038    A
DBG=IDP   BUB3     SB7    B7-B1                                                    DBG=IDP    3039    A
DBG=IDP            SA5    A5+B1                                                    DBG=IDP    3040    A
DBG=IDP            LE     B7,B0,BUB2C IF NO MORE SOURCE WORDS TO BURST/BUILD       DBG=IDP    3041    A
DBG=IDP            IX7    X5-X1                                                    DBG=IDP    3042    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            SB6    A4                                                    DBG=IDP    3043    A
DBG=IDP            ZR     X7,BUB3     IF NEXT SOURCE IMAGE WORD IS ALL BLANK    DBG=IDP    3044    A
DBG=IDP                                                                         DBG=IDP    3045    A
DBG=IDP   BUB4     LX5    CHAR                                                  DBG=IDP    3046    A
DBG=IDP            LE     B6,B0,BUB3  IF SOURCE IMAGE WORD (X5) IS EXHAUSTED    DBG=IDP    3047    A
DBG=IDP            BX4    -X0*X5                                                DBG=IDP    3048    A
DBG=IDP            SB6    B6-B3                                                 DBG=IDP    3049    A
DBG=IDP            SB2    X4+7777B-1RM                                          DBG=IDP    3050    A
DBG=IDP            NO                                                           DBG=IDP    3051    A
DBG=IDP            LX7    X2,B2                                                 DBG=IDP    3052    A
DBG=IDP            EQ     B2,B5,BUB4  IF CHAR TO PACK IS A BLANK (55B)          DBG=IDP    3053    A
DBG=IDP            LX3    X4,B4                                                 DBG=IDP    3054    A
DBG=IDP            SB4    B4-B3                                                 DBG=IDP    3055    A
DBG=IDP            MI     X7,BUB5     IF NEXT CHAR TO PACK IS TERMINATOR        DBG=IDP    3056    A
DBG=IDP            BX6    X6+X3                                                 DBG=IDP    3057    A
DBG=IDP            GE     B4,B0,BUB4  IF PACKING REG (X6) NOT FULL              DBG=IDP    3058    A
DBG=IDP                                                                         DBG=IDP    3059    A
DBG=IDP   *        HERE IF AN ERROR -- TERMINATE TOKEN BUFFER.                  DBG=IDP    3060    A
DBG=IDP                                                                         DBG=IDP    3061    A
DBG=IDP            SA1    IDPSTMT                                               DBG=IDP    3062    A
DBG=IDP            MX6    1                                                     DBG=IDP    3063    A
DBG=IDP            SX2    B1                                                    DBG=IDP    3064    A
DBG=IDP            BX6    X6+X2       (X6) = 1/1=LAST STMT ON LINE,41/0,18/LEN  DBG=IDP    3065    A
DBG=IDP            SA6    X1                                                    DBG=IDP    3066    A
DBG=IDP            EQ     IDP=ER      ERROR - TOKEN .GT. 7 CHARS                DBG=IDP    3067    A
DBG=IDP                                                                         DBG=IDP    3068    A
DBG=IDP   *        HERE TO STORE PACKING REG (X6).                              DBG=IDP    3069    A
DBG=IDP                                                                         DBG=IDP    3070    A
DBG=IDP   BUB5     LX6    10*CHAR-8*CHAR (X6) = 42/0LNAME,18/0                  DBG=IDP    3071    A
DBG=IDP            SA1    BUB=LEN                                               DBG=IDP    3072    A
DBG=IDP            SA0    B1                                                    DBG=IDP    3073    A
DBG=IDP            SA2    MX=NR09                                               DBG=IDP    3074    A
DBG=IDP            LX7    X6,B3                                                 DBG=IDP    3075    A
DBG=IDP            BX7    -X0*X7                                                DBG=IDP    3076    A
DBG=IDP            SX3    X1-3-1                                                DBG=IDP    3077    A
DBG=IDP            SB4    X7                                                    DBG=IDP    3078    A
DBG=IDP            LX2    X2,B4                                                 DBG=IDP    3079    A
DBG=IDP            PL     X3,BUB6     IF NO TABLE OVERFLOW IMPENDING            DBG=IDP    3080    A
DBG=IDP                                                                         DBG=IDP    3081    A
DBG=IDP   *        HERE IF AN ERROR -- TERMINATE TOKEN BUFFER.                  DBG=IDP    3082    A
DBG=IDP                                                                         DBG=IDP    3083    A
DBG=IDP            SA1    IDPSTMT                                               DBG=IDP    3084    A
DBG=IDP            MX6    1                                                     DBG=IDP    3085    A
DBG=IDP            SX2    B1                                                    DBG=IDP    3086    A
DBG=IDP            BX6    X6+X2       (X6) = 1/1=LAST STMT ON LINE,41/0,18/LEN  DBG=IDP    3087    A
DBG=IDP            SA6    X1                                                    DBG=IDP    3088    A
DBG=IDP            EQ     IDP=OV                                                DBG=IDP    3089    A
DBG=IDP                                                                         DBG=IDP    3090    A
DBG=IDP   BUB6     ZR     X6,BUB8     IF (X6) IS NULL, PROBABLE ,, SYNTAX       DBG=IDP    3091    A
DBG=IDP            SX7    O.VAR                                                 DBG=IDP    3092    A
DBG=IDP            PL     X2,BUB7     IF 1ST CHAR IN (X6) IS NOT 0 THRU 9       DBG=IDP    3093    A
DBG=IDP            SX7    O.CONS                                                DBG=IDP    3094    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP                                                                    DBG=IDP    3095    A
DBG=IDP   BUB7     BX6    X6+X7                                            DBG=IDP    3096    A
DBG=IDP            SA0    A0+B1                                            DBG=IDP    3097    A
DBG=IDP            SA6    A6+B1                                            DBG=IDP    3098    A
DBG=IDP                                                                    DBG=IDP    3099    A
DBG=IDP   *        CHECK FOR AND PROCESS END-OF-STATEMENT.                 DBG=IDP    3100    A
DBG=IDP                                                                    DBG=IDP    3101    A
DBG=IDP   BUB8     SB2    -2                                               DBG=IDP    3102    A
DBG=IDP            SX2    X4-1R"EOS"                                       DBG=IDP    3103    A
DBG=IDP            LE     B7,B2,BUB9  IF NOTHING MORE TO BURST             DBG=IDP    3104    A
DBG=IDP            NZ     X2,BUB10    IF NOT EOS CHARACTER                 DBG=IDP    3105    A
DBG=IDP                                                                    DBG=IDP    3106    A
DBG=IDP   BUB9     SA1    IDPSTMT                                          DBG=IDP    3107    A
DBG=IDP            MX6    0                                                DBG=IDP    3108    A
DBG=IDP            SX2    A6+B1                                            DBG=IDP    3109    A
DBG=IDP            SA6    A6+B1       MARK EOS                             DBG=IDP    3110    A
DBG=IDP            IX7    X2-X1                                            DBG=IDP    3111    A
DBG=IDP            NO                                                      DBG=IDP    3112    A
DBG=IDP            SA7    X1                                               DBG=IDP    3113    A
DBG=IDP            LE     B7,B2,BUB11 IF FINISHED                          DBG=IDP    3114    A
DBG=IDP            SX7    A6+B1       (X7) = ADDR OF LEN WORD FOR NEXT STMT DBG=IDP   3115    A
DBG=IDP            SA6    A6+B1       (A6+1) = NEXT ADDR TO STORE          DBG=IDP    3116    A
DBG=IDP            SA0    A0+B1                                            DBG=IDP    3117    A
DBG=IDP            SA7    A1                                               DBG=IDP    3118    A
DBG=IDP            EQ     BUB2        CRANK UP FOR NEW STMT...             DBG=IDP    3119    A
DBG=IDP                                                                    DBG=IDP    3120    A
DBG=IDP   *        HERE TO STORE TERMINATOR CHARACTER (X4).                DBG=IDP    3121    A
DBG=IDP                                                                    DBG=IDP    3122    A
DBG=IDP   BUB10    BX6    X4                                               DBG=IDP    3123    A
DBG=IDP            SX7    O.SEP                                            DBG=IDP    3124    A
DBG=IDP            LX6    10*CHAR-1*CHAR                                   DBG=IDP    3125    A
DBG=IDP            BX6    X6+X7       (X6) = 42/0LCHAR,18/O.SEP            DBG=IDP    3126    A
DBG=IDP            SA6    A6+B1                                            DBG=IDP    3127    A
DBG=IDP            EQ     BUB2A       CONTINUE...                          DBG=IDP    3128    A
DBG=IDP                                                                    DBG=IDP    3129    A
DBG=IDP   BUB11    MX6    1                                                DBG=IDP    3130    A
DBG=IDP            BX7    X6+X7                                            DBG=IDP    3131    A
DBG=IDP            SA7    X1                                               DBG=IDP    3132    A
DBG=IDP            EQ     EXIT.                                            DBG=IDP    3133    A
DBG=IDP                                                                    DBG=IDP    3134    A
DBG=IDP                                                                    DBG=IDP    3135    A
DBG=IDP   BUB=LEN  =      APL         SAVED NR OF WORDS LEFT IN TOKEN BUFFER DBG=IDP 3136    A
DBG=IDP   CBC      SPACE  4,8                                              DBG=IDP    3137    A
DBG=IDP   **       CBC - CHECK BREAK CONDITION.                            DBG=IDP    3138    A
DBG=IDP   *                                                                DBG=IDP    3139    A
DBG=IDP   *                                                                DBG=IDP    3140    A
DBG=IDP   *               THIS ROUTINE RETURNS A TRUE/FALSE VALUE FOR A GIVEN DBG=IDP 3141  A
DBG=IDP   *        LOGICAL EXPRESSION-- C1.OP.C2                           DBG=IDP    3142    A
DBG=IDP   *                                                                DBG=IDP    3143    A
DBG=IDP   *        NOTE-- THE OPERATOR JUMP TABLE AT *CBC.OP* HAS A ONE-TO-ONE DBG=IDP 3144  A
DBG=IDP   *        ORDER DEPENDENT RELATIONSHIP WITH THE *BREAK* KEYWORDS AT DBG=IDP 3145    A
DBG=IDP   *        (IDPKBR) ET SEQ. SHOULD A CHANGE BE EFFECTED TO EITHER TABLE, DBG=IDP 3146  A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *         BOTH SHOULD BE CHECKED FOR CONSISTENCY.                    DBG=IDP    3147    A
DBG=IDP  *                                                                    DBG=IDP    3148    A
DBG=IDP  *         ENTRY  (X1) = C1   1ST QUANTITY                            DBG=IDP    3149    A
DBG=IDP  *                (X2) = C2   2ND QUANTITY                            DBG=IDP    3150    A
DBG=IDP  *                (B5) = BREAK TYPE CODE (BTC)                        DBG=IDP    3151    A
DBG=IDP  *                                                                    DBG=IDP    3152    A
DBG=IDP  *         EXIT   (B6) = .NZ. IF TRUE, ELSE .ZR. IF FALSE             DBG=IDP    3153    A
DBG=IDP  *                                                                    DBG=IDP    3154    A
DBG=IDP  *         USES   X - 0,3,6,7                                         DBG=IDP    3155    A
DBG=IDP  *                A - NONE                                            DBG=IDP    3156    A
DBG=IDP  *                B - 6                                               DBG=IDP    3157    A
DBG=IDP  *                                                                    DBG=IDP    3158    A
DBG=IDP  *         CALLS  NONE                                                DBG=IDP    3159    A
DBG=IDP                                                                       DBG=IDP    3160    A
DBG=IDP                                                                       DBG=IDP    3161    A
DBG=IDP   CBC      SUBR                ** ENTRY/EXIT **                       DBG=IDP    3162    A
DBG=IDP            MI     B5,*+4S15    IF BTC IS BAD, AVOID SMELLY EVIL BUG...DBG=IDP    3163    A
DBG=IDP            SB6    CBC2-CBC.OP (B6) = NR OF OPERATOR JP TABLE ENTRIES  DBG=IDP    3164    A
DBG=IDP            GT     B5,B6,*+4S15  IF BTC IS BAD, P U...DUMMKOPF         DBG=IDP    3165    A
DBG=IDP            SB6    B1           SET TO *CONDITION SATISFIED (TRUE)*    DBG=IDP    3166    A
DBG=IDP            ZR     B5,EXIT.     IF UNCONDITIONAL BREAK                 DBG=IDP    3167    A
DBG=IDP            BX3    X1-X2                                               DBG=IDP    3168    A
DBG=IDP            IX6    X1-X2                                               DBG=IDP    3169    A
DBG=IDP            IX7    X2-X1                                               DBG=IDP    3170    A
DBG=IDP            MX0    0                                                   DBG=IDP    3171    A
DBG=IDP            IX6    X6+X0        REMOVE POSSIBLE -0                     DBG=IDP    3172    A
DBG=IDP            IX7    X7+X0                                               DBG=IDP    3173    A
DBG=IDP            JP     CBC.OP+B5-1                                         DBG=IDP    3174    A
DBG=IDP                                                                       DBG=IDP    3175    A
DBG=IDP  *         CBC.OP - OPERATOR JUMP TABLE.                              DBG=IDP    3176    A
DBG=IDP                                                                       DBG=IDP    3177    A
DBG=IDP   CBC.OP   BSS    0                                                   DBG=IDP    3178    A
DBG=IDP            LOC    1                                                   DBG=IDP    3179    A
DBG=IDP                                                                       DBG=IDP    3180    A
DBG=IDP  *         HERE FOR .PL.C1                                            DBG=IDP    3181    A
DBG=IDP                                                                       DBG=IDP    3182    A
DBG=IDP   +        PL     X1,EXIT.     IF TRUE                                DBG=IDP    3183    A
DBG=IDP            EQ     CBC2                                                DBG=IDP    3184    A
DBG=IDP                                                                       DBG=IDP    3185    A
DBG=IDP  *         HERE FOR .MI.C1                                            DBG=IDP    3186    A
DBG=IDP                                                                       DBG=IDP    3187    A
DBG=IDP  +         MI     X1,EXIT.     IF TRUE                                DBG=IDP    3188    A
DBG=IDP            EQ     CBC2                                                DBG=IDP    3189    A
DBG=IDP                                                                       DBG=IDP    3190    A
DBG=IDP  *         HERE FOR .ZR.C1                                            DBG=IDP    3191    A
DBG=IDP                                                                       DBG=IDP    3192    A
DBG=IDP   +        ZR     X1,EXIT.     IF TRUE                                DBG=IDP    3193    A
DBG=IDP            EQ     CBC2                                                DBG=IDP    3194    A
DBG=IDP                                                                       DBG=IDP    3195    A
DBG=IDP  *         HERE FOR .NZ.C1                                            DBG=IDP    3196    A
DBG=IDP                                                                       DBG=IDP    3197    A
DBG=IDP   +        NZ     X1,EXIT.     IF TRUE                                DBG=IDP    3198    A
```

```
         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            EQ     CBC2                                                    DBG=IDP     3199     A
DBG=IDP                                                                            DBG=IDP     3200     A
DBG=IDP  *         HERE FOR C1.EQ.C2                                               DBG=IDP     3201     A
DBG=IDP                                                                            DBG=IDP     3202     A
DBG=IDP   +        ZR     X3,EXIT.    IF TRUE                                      DBG=IDP     3203     A
DBG=IDP            EQ     CBC2                                                    DBG=IDP     3204     A
DBG=IDP                                                                            DBG=IDP     3205     A
DBG=IDP  *         HERE FOR C1.NE.C2                                               DBG=IDP     3206     A
DBG=IDP                                                                            DBG=IDP     3207     A
DBG=IDP   +        NZ     X3,EXIT.    IF TRUE                                      DBG=IDP     3208     A
DBG=IDP            EQ     CBC2                                                    DBG=IDP     3209     A
DBG=IDP                                                                            DBG=IDP     3210     A
DBG=IDP  *         HERE FOR C1.LT.C2                                               DBG=IDP     3211     A
DBG=IDP                                                                            DBG=IDP     3212     A
DBG=IDP   +        MI     X6,EXIT.    IF TRUE                                      DBG=IDP     3213     A
DBG=IDP            EQ     CBC2                                                    DBG=IDP     3214     A
DBG=IDP                                                                            DBG=IDP     3215     A
DBG=IDP  *         HERE FOR C1.GE.C2                                               DBG=IDP     3216     A
DBG=IDP                                                                            DBG=IDP     3217     A
DBG=IDP   +        PL     X6,EXIT.    IF TRUE                                      DBG=IDP     3218     A
DBG=IDP            EQ     CBC2                                                    DBG=IDP     3219     A
DBG=IDP                                                                            DBG=IDP     3220     A
DBG=IDP  *         HERE FOR C1.LE.C2                                               DBG=IDP     3221     A
DBG=IDP                                                                            DBG=IDP     3222     A
DBG=IDP   +        PL     X7,EXIT.    IF TRUE                                      DBG=IDP     3223     A
DBG=IDP            EQ     CBC2                                                    DBG=IDP     3224     A
DBG=IDP                                                                            DBG=IDP     3225     A
DBG=IDP  *         HERE FOR C1.GT.C2                                               DBG=IDP     3226     A
DBG=IDP                                                                            DBG=IDP     3227     A
DBG=IDP   +        MI     X7,EXIT.    IF TRUE                                      DBG=IDP     3228     A
DBG=IDP            EQ     CBC2                                                    DBG=IDP     3229     A
DBG=IDP                                                                            DBG=IDP     3230     A
DBG=IDP            LOC    *O                                                      DBG=IDP     3231     A
DBG=IDP                                                                            DBG=IDP     3232     A
DBG=IDP  *         HERE FOR CONDITION NOT SATISFIED (FALSE).                       DBG=IDP     3233     A
DBG=IDP                                                                            DBG=IDP     3234     A
DBG=IDP   CBC2     SB6    B0                                                      DBG=IDP     3235     A
DBG=IDP            EQ     EXIT.                                                   DBG=IDP     3236     A
DBG=IDP   CLZ      SPACE  4,8                                                     DBG=IDP     3237     A
DBG=IDP  **        CLZ - CLEAR *IDP* TABLE.                                       DBG=IDP     3238     A
DBG=IDP  *                                                                         DBG=IDP     3239     A
DBG=IDP  *                                                                         DBG=IDP     3240     A
DBG=IDP  *                THIS ROUTINE WILL CLEAR AN *IDP* ZERO TERMINATED TABLE   DBG=IDP     3241     A
DBG=IDP  *         BY SETTING ALL ENTRIES TO *AVAILABLE*, I.E. -1.                 DBG=IDP     3242     A
DBG=IDP  *                                                                         DBG=IDP     3243     A
DBG=IDP  *         ENTRY  (B6) = FWA OF TABLE TO BE CLEARED                        DBG=IDP     3244     A
DBG=IDP  *                                                                         DBG=IDP     3245     A
DBG=IDP  *         EXIT   NONE                                                     DBG=IDP     3246     A
DBG=IDP  *                                                                         DBG=IDP     3247     A
DBG=IDP  *         USES   X - 1,6                                                  DBG=IDP     3248     A
DBG=IDP  *                A - 1,6                                                  DBG=IDP     3249     A
DBG=IDP  *                B - NONE                                                 DBG=IDP     3250     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *                                                                  DBG=IDP   3251   A
DBG=IDP  *         CALLS  NONE                                              DBG=IDP   3252   A
DBG=IDP                                                                     DBG=IDP   3253   A
DBG=IDP                                                                     DBG=IDP   3254   A
DBG=IDP   CLZ      SUBR              ** ENTRY/EXIT **                       DBG=IDP   3255   A
DBG=IDP            SA1    B6                                                 DBG=IDP   3256   A
DBG=IDP            MX6    -1                                                 DBG=IDP   3257   A
DBG=IDP                                                                     DBG=IDP   3258   A
DBG=IDP   CLZ2     ZR     X1,EXIT.    IF FINISHED                            DBG=IDP   3259   A
DBG=IDP            SA6    A1                                                 DBG=IDP   3260   A
DBG=IDP            SA1    A1+B1                                              DBG=IDP   3261   A
DBG=IDP            EQ     CLZ2                                               DBG=IDP   3262   A
DBG=IDP   CON      SPACE  4,8                                                DBG=IDP   3263   A
DBG=IDP  **        CON - CONNECT/DISCONT FILE.                              DBG=IDP   3264   A
DBG=IDP  *                                                                  DBG=IDP   3265   A
DBG=IDP  *                                                                  DBG=IDP   3266   A
DBG=IDP  *         ENTRY  (A1,X1) = A + C OF 1ST WORD OF FET OF FILE        DBG=IDP   3267   A
DBG=IDP  *                (X2)    = +0  IF TO CONNECT                       DBG=IDP   3268   A
DBG=IDP  *                        =  1  IF TO DISCONT                       DBG=IDP   3269   A
DBG=IDP  *                                                                  DBG=IDP   3270   A
DBG=IDP  *         EXIT   SCOPE 3--                                         DBG=IDP   3271   A
DBG=IDP  *                (X1)    = .MI. IF PP ROUTINE *CON* DETECTED ERROR, DBG=IDP   3272   A
DBG=IDP  *                                ELSE .PL.                         DBG=IDP   3273   A
DBG=IDP  *                KRONOS/NOS/SCOPE 2-- NONE                         DBG=IDP   3274   A
DBG=IDP  *                                                                  DBG=IDP   3275   A
DBG=IDP  *         USES   ALL BUT A0,A5,X5                                  DBG=IDP   3276   A
DBG=IDP  *                                                                  DBG=IDP   3277   A
DBG=IDP  *         CALLS  SCOPE 3-- SYSTEM                                  DBG=IDP   3278   A
DBG=IDP  *                KRONOS/NOS-- CLOSE,REQUEST,WRITER                 DBG=IDP   3279   A
DBG=IDP                                                                     DBG=IDP   3280   A
DBG=IDP                                                                     DBG=IDP   3281   A
DBG=IDP   CON      SUBR              ** ENTRY/EXIT **                       DBG=IDP   3282   A
DBG=IDP                                                                     DBG=IDP   3283   A
DBG=IDP   #OS2     IFEQ   .OS,2       IF SCOPE 2                            DBG=IDP   3284   A
DBG=IDP            EQ     EXIT.                                              DBG=IDP   3285   A
DBG=IDP   #OS2     ENDIF                                                     DBG=IDP   3286   A
DBG=IDP                                                                     DBG=IDP   3287   A
DBG=IDP   #OS3     IFEQ   .OS,3       IF SCOPE 3                            DBG=IDP   3288   A
DBG=IDP            MX0    7*CHAR                                             DBG=IDP   3289   A
DBG=IDP            BX6    X0*X1       (X6) = 42/0LFILENAME, 18/0             DBG=IDP   3290   A
DBG=IDP            SA6    CONA                                               DBG=IDP   3291   A
DBG=IDP            LX2    6                                                  DBG=IDP   3292   A
DBG=IDP            SYSTEM CON,RCL,CONA,X2                                    DBG=IDP   3293   A
DBG=IDP                                                                     DBG=IDP   3294   A
DBG=IDP            SA1    CONA                                               DBG=IDP   3295   A
DBG=IDP            LX1    59-1                                               DBG=IDP   3296   A
DBG=IDP            EQ     EXIT.                                              DBG=IDP   3297   A
DBG=IDP                                                                     DBG=IDP   3298   A
DBG=IDP                                                                     DBG=IDP   3299   A
DBG=IDP   CONA     BSSZ   1           PARAMETER WORD FOR PP ROUTINE *CON*--  DBG=IDP   3300   A
DBG=IDP  *                              42/0LFILENAME                       DBG=IDP   3301   A
DBG=IDP  *                              16/0, 1/ERROR FLAG, 1/COMPLETE BIT  DBG=IDP   3302   A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP                                                                                  DBG=IDP     3303     A
DBG=IDP   #OS3     ELSE                IF KRONOS/NOS                                     DBG=IDP     3304     A
DBG=IDP            SX6    A1                                                             DBG=IDP     3305     A
DBG=IDP            SA6    CONA         SAVE (A1) = FWA OF FET                            DBG=IDP     3306     A
DBG=IDP            NZ     X2,CON2      IF TO DISCONT FILE                                DBG=IDP     3307     A
DBG=IDP                                                                                  DBG=IDP     3308     A
DBG=IDP   *        HERE TO CONNECT FILE.                                                 DBG=IDP     3309     A
DBG=IDP                                                                                  DBG=IDP     3310     A
DBG=IDP            WRITER A1,,RCL      FLUSH BUFFER                                      DBG=IDP     3311     A
DBG=IDP            SA1    CONA                                                           DBG=IDP     3312     A
DBG=IDP            CLOSE  X1,,RCL      CLOSE FILE                                        DBG=IDP     3313     A
DBG=IDP            SA1    CONA                                                           DBG=IDP     3314     A
DBG=IDP            SX6    2RTT                                                           DBG=IDP     3315     A
DBG=IDP            MX0    60-48                                                          DBG=IDP     3316     A
DBG=IDP            SA2    X1+B1        (X2) = 12/DEVICE TYPE, 48/OTHER                   DBG=IDP     3317     A
DBG=IDP            LX6    10*CHAR-2*CHAR                                                 DBG=IDP     3318     A
DBG=IDP            BX2    -X0*X2                                                         DBG=IDP     3319     A
DBG=IDP            IX6    X2+X6        (X6) = 12/2RTT, 48/OTHER                          DBG=IDP     3320     A
DBG=IDP            SA6    A2                                                             DBG=IDP     3321     A
DBG=IDP            REQUEST X1,TT       ASSIGN FILE TO TERMINAL                           DBG=IDP     3322     A
DBG=IDP            EQ     EXIT.                                                          DBG=IDP     3323     A
DBG=IDP                                                                                  DBG=IDP     3324     A
DBG=IDP   *        HERE TO DISCONT FILE.                                                 DBG=IDP     3325     A
DBG=IDP                                                                                  DBG=IDP     3326     A
DBG=IDP    CON2    WRITER A1,,RCL      FLUSH BUFFER                                      DBG=IDP     3327     A
DBG=IDP            SA1    CONA                                                           DBG=IDP     3328     A
DBG=IDP            CLOSE  X1,,RCL                                                        DBG=IDP     3329     A
DBG=IDP            EQ     EXIT.                                                          DBG=IDP     3330     A
DBG=IDP                                                                                  DBG=IDP     3331     A
DBG=IDP                                                                                  DBG=IDP     3332     A
DBG=IDP    CONA    BSSZ   1            SAVED FWA OF FET                                  DBG=IDP     3333     A
DBG=IDP    #OS3    ENDIF                                                                 DBG=IDP     3334     A
DBG=IDP    DAZ     SPACE  4,8                                                            DBG=IDP     3335     A
DBG=IDP   **       DAZ - DEASSEMBLER.                                                    DBG=IDP     3336     A
DBG=IDP   *                                                                              DBG=IDP     3337     A
DBG=IDP   *                                                                              DBG=IDP     3338     A
DBG=IDP   *                DEASSEMBLES A SINGLE BINARY INSTRUCTION TO ITS                DBG=IDP     3339     A
DBG=IDP   *        *COMPASS* MNEMONIC AND LISTS IT.                                      DBG=IDP     3340     A
DBG=IDP   *                                                                              DBG=IDP     3341     A
DBG=IDP   *        ENTRY  (A5,     =  ADDR OF WORD CONTAINING INSTRUCTION TO            DBG=IDP     3342     A
DBG=IDP   *                             DEASSEMBLE                                       DBG=IDP     3343     A
DBG=IDP   *                   X5)  =  WORD CONTAINING INSTRUCTION TO DEASSEMBLE          DBG=IDP     3344     A
DBG=IDP   *                           NOTE-- ((A5)) MIGHT NOT BE EQUAL TO (X5),          DBG=IDP     3345     A
DBG=IDP   *                             AS WOULD BE THE CASE WHEN DASSEMBLING            DBG=IDP     3346     A
DBG=IDP   *                             THE SAVED PREVIOUS CONTENTS OF AN IDP            DBG=IDP     3347     A
DBG=IDP   *                             GENERATED BREAK (SEE *STP*)                      DBG=IDP     3348     A
DBG=IDP   *                (B4)     =  POSITION COUNTER. LEFT MOST BIT TO BE             DBG=IDP     3349     A
DBG=IDP   *                             DEASSEMBLED IS BIT (B4)-1 IN (X5).               DBG=IDP     3350     A
DBG=IDP   *                             (BITS ARE NUMBERED 59 THRU 0) COMPASS            DBG=IDP     3351     A
DBG=IDP   *                             EQUIVALENT WOULD BE--                            DBG=IDP     3352     A
DBG=IDP   *                                 POS   (B4)    IN (X5)                        DBG=IDP     3353     A
DBG=IDP   *                (IDPFAD) =  .NZ. IF *DAZ* IS TO UNCONDITIONALLY FORCE        DBG=IDP     3354     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *                                OUT ADDR OF DEASSEMBLED INSTRUCTION,          DBG=IDP    3355    A
DBG=IDP  *                        =  .ZR. IF ADDR IS TO APPEAR ONLY IF INSTR IS          DBG=IDP    3356    A
DBG=IDP  *                                FORCED UPPER IN WORD, I.E. (B4)=60D            DBG=IDP    3357    A
DBG=IDP  *                                                                               DBG=IDP    3358    A
DBG=IDP  *        EXIT   (A5,X5)  =  UNCHANGED                                           DBG=IDP    3359    A
DBG=IDP  *               (B4)     =  UNCHANGED                                           DBG=IDP    3360    A
DBG=IDP  *               (X1)     =  .MI. IF INSTRUCTION DEASSEMBLED FORCES              DBG=IDP    3361    A
DBG=IDP  *                            UPPER, ELSE .PL.                                   DBG=IDP    3362    A
DBG=IDP  *               (B5)     =  NR OF BITS DEASSEMBLED IF NO FORCE UPPER,           DBG=IDP    3363    A
DBG=IDP  *                        =  NR OF BITS IN INSTRUCTION THAT FORCED UPPER         DBG=IDP    3364    A
DBG=IDP  *               (IDPFAD) =  +0                                                  DBG=IDP    3365    A
DBG=IDP  *                                                                               DBG=IDP    3366    A
DBG=IDP  *        USES   ALL BUT A5,X5,B4                                                DBG=IDP    3367    A
DBG=IDP  *                                                                               DBG=IDP    3368    A
DBG=IDP  *        LOCKED (A6+1)    =  NEXT AVAILABLE *SNAPLNE* LOCATION                  DBG=IDP    3369    A
DBG=IDP  *                                                                               DBG=IDP    3370    A
DBG=IDP  *        CALLS  CAD,PRIDP,VFD                                     000930        DBG=IDP    3371    A
DBG=IDP                                                                                  DBG=IDP    3372    A
DBG=IDP                                                                                  DBG=IDP    3373    A
DBG=IDP   DAZ     SUBR                ** ENTRY/EXIT **                                   DBG=IDP    3374    A
DBG=IDP           SB6    60                                                              DBG=IDP    3375    A
DBG=IDP           LE     B4,B0,*+1S17  IF POSITION COUNTER IS BAD                        DBG=IDP    3376    A
DBG=IDP           SA1    =10H                                                            DBG=IDP    3377    A
DBG=IDP           SA2    IDPFAD                                                          DBG=IDP    3378    A
DBG=IDP           MX7    0                                                               DBG=IDP    3379    A
DBG=IDP           BX6    X1                                                              DBG=IDP    3380    A
DBG=IDP           SA7    A2            CLEAR *FORCE OUT ADDR UNCONDITIONALLY* FLAG       DBG=IDP    3381    A
DBG=IDP           NZ     X2,DAZ1       IF FORCING OUT ADDR UNCONDITIONALLY               DBG=IDP    3382    A
DBG=IDP           LT     B4,B6,DAZ2    IF THIS INSTRUCTION NOT UPPER                     DBG=IDP    3383    A
DBG=IDP                                                                                  DBG=IDP    3384    A
DBG=IDP  *        HERE IF INSTRUCTION IS UPPER, OUTPUT ADDRESS.                          DBG=IDP    3385    A
DBG=IDP                                                                                  DBG=IDP    3386    A
DBG=IDP   DAZ1    SX0    B4            SAVE (B4)                                 000950  DBG=IDP    3387    A
DBG=IDP           SX1    A5            (X1) = ADDR OF INSTRUCTION TO DEASSEMBLE  000960  DBG=IDP    3388    A
DBG=IDP           RJ     CAD           CONVERT ADDR TO DPC                       000970  DBG=IDP    3389    A
DBG=IDP           SB4    X0            RESTORE (B4)                              000980  DBG=IDP    3390    A
DBG=IDP                                                                                  DBG=IDP    3391    A
DBG=IDP  *        BREAK OUT OP CODE.                                                     DBG=IDP    3392    A
DBG=IDP                                                                                  DBG=IDP    3393    A
DBG=IDP   DAZ2    SA6    SNAPLNE                                                         DBG=IDP    3394    A
DBG=IDP           SB6    -B4                                                             DBG=IDP    3395    A
DBG=IDP           MX0    -6                                                              DBG=IDP    3396    A
DBG=IDP           SB6    60+B6+6       60-(B4) = LEFT JUST SHIFT COUNT FOR INSTRTN       DBG=IDP    3397    A
DBG=IDP           LX5    X5,B6         (X5) = 54/OTHER STUFF, 6/OP CODE                  DBG=IDP    3398    A
DBG=IDP           BX7    -X0*X5                                                          DBG=IDP    3399    A
DBG=IDP           LX5    3             (X5) = 51/OTHER STUFF,6/OP CODE,3/I               DBG=IDP    3400    A
DBG=IDP           MX0    -3                                                              DBG=IDP    3401    A
DBG=IDP           SA7    DAZ=OP                                                          DBG=IDP    3402    A
DBG=IDP           ZR     X7,DAZ12      IF 00B OP CODE, REST OF WORD IS DATA              DBG=IDP    3403    A
DBG=IDP                                                                                  DBG=IDP    3404    A
DBG=IDP  *        BREAK OUT *I* FIELD.                                                   DBG=IDP    3405    A
DBG=IDP                                                                                  DBG=IDP    3406    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP          BX7    -X0*X5                                          DBG=IDP    3407    A
DBG=IDP          LX5    3          (X5) = 48/OTHER STUFF,6/OP,3/I,3/J   DBG=IDP    3408    A
DBG=IDP          SX7    X7+1R0                                          DBG=IDP    3409    A
DBG=IDP          SA7    A7+B1      SAVE (X7) = 1R<I>                    DBG=IDP    3410    A
DBG=IDP                                                                 DBG=IDP    3411    A
DBG=IDP  *       BREAK OUT *J* FIELD.                                   DBG=IDP    3412    A
DBG=IDP                                                                 DBG=IDP    3413    A
DBG=IDP          BX7    -X0*X5                                          DBG=IDP    3414    A
DBG=IDP          LX5    3          (X5) = 45/OTHER,6/OP,3/I,3/J,3/K     DBG=IDP    3415    A
DBG=IDP          SX7    X7+1R0                                          DBG=IDP    3416    A
DBG=IDP          SA7    A7+B1      SAVE (X7) = 1R<J>                    DBG=IDP    3417    A
DBG=IDP                                                                 DBG=IDP    3418    A
DBG=IDP  *       BREAK OUT *K* FIELD.                                   DBG=IDP    3419    A
DBG=IDP                                                                 DBG=IDP    3420    A
DBG=IDP          BX7    -X0*X5                                          DBG=IDP    3421    A
DBG=IDP          LX5    18-3       (X5) = 30/OTHER,6/OP,3/I,3/J,18/Q    DBG=IDP    3422    A
DBG=IDP          SX7    X7+1R0                                          DBG=IDP    3423    A
DBG=IDP          MX0    -18                                             DBG=IDP    3424    A
DBG=IDP          SA7    A7+B1      SAVE (X7) = 1R<K>                    DBG=IDP    3425    A
DBG=IDP                                                                 DBG=IDP    3426    A
DBG=IDP  *       BREAK OUT *Q* 18 BIT ADDRESS FIELD.                    DBG=IDP    3427    A
DBG=IDP                                                                 DBG=IDP    3428    A
DBG=IDP          BX1    -X0*X5     (X1) = 42/0,18/Q                     DBG=IDP    3429    A
DBG=IDP          SB5    B4         SAVE (B4)                            DBG=IDP    3430    A
DBG=IDP          SB7    60-60+B4-6-3-3-3-15  (B7) = 60-((60-(B4))+6+3+3+3+15)  DBG=IDP    3431    A
DBG=IDP          SX7    X1                                              DBG=IDP    3432    A
DBG=IDP          LX5    X5,B7      RESTORE (X5)                         DBG=IDP    3433    A
DBG=IDP          SA7    DAZ=ADR                                         DBG=IDP    3434    A
DBG=IDP          RJ     =XCOD      CONVERT BINARY TO OCTAL DPC-- *Q*    DBG=IDP    3435    A
DBG=IDP          SB4    B5         RESTORE (B4)                         DBG=IDP    3436    A
DBG=IDP          SB2    B2-B1                                           DBG=IDP    3437    A
DBG=IDP          MX0    1                                               DBG=IDP    3438    A
DBG=IDP          AX0    X0,B2                                           DBG=IDP    3439    A
DBG=IDP          BX7    X0*X4      (X7) = 0L<Q>                         DBG=IDP    3440    A
DBG=IDP          SA7    A7+B1                                           DBG=IDP    3441    A
DBG=IDP                                                                 DBG=IDP    3442    A
DBG=IDP  *       FETCH INSTRUCTION SKELETON AND CHECK FOR LENGTH CONFLICT.  DBG=IDP    3443    A
DBG=IDP                                                                 DBG=IDP    3444    A
DBG=IDP          SA1    DAZ=OP                                          DBG=IDP    3445    A
DBG=IDP          SA2    DAZ=PS+X1  (X2) = INSTRUCTION SKELETON          DBG=IDP    3446    A
DBG=IDP          SB5    15                                              DBG=IDP    3447    A
DBG=IDP          BX1    X5                                              DBG=IDP    3448    A
DBG=IDP          PL     X2,DAZ3    IF SHORT (15 BIT) INSTRUCTION        DBG=IDP    3449    A
DBG=IDP          SB5    B5+B5      (B5) = 30                            DBG=IDP    3450    A
DBG=IDP                                                                 DBG=IDP    3451    A
DBG=IDP   DAZ3   LT     B4,B5,DAZ12 IF NOT ENOUGH ROOM FOR 30 BIT INSTR,  DBG=IDP    3452    A
DBG=IDP  *                              MUST BE DATA                    DBG=IDP    3453    A
DBG=IDP                                                                 DBG=IDP    3454    A
DBG=IDP  *       CONVERT BINARY INSTRUCTION PARCEL TO OCTAL DPC.        DBG=IDP    3455    A
DBG=IDP                                                                 DBG=IDP    3456    A
DBG=IDP          RJ     VFD        VARIABLE FIELD DEFINITION            DBG=IDP    3457    A
DBG=IDP          SA6    A6+B1                                           DBG=IDP    3458    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            BX6    X7                                                    DBG=IDP    3459    A
DBG=IDP            SA6    A6+B1                                                 DBG=IDP    3460    A
DBG=IDP                                                                         DBG=IDP    3461    A
DBG=IDP  *         CHECK FOR X REGISTER BRANCH INSTRUCTION.                     DBG=IDP    3462    A
DBG=IDP                                                                         DBG=IDP    3463    A
DBG=IDP            SA1    DAZ=OP                                                DBG=IDP    3464    A
DBG=IDP            SA2    DAZ=PS+X1    (X2) = INSTRUCTION SKELETON              DBG=IDP    3465    A
DBG=IDP            SX6    X1-03B                                                DBG=IDP    3466    A
DBG=IDP            NZ     X6,DAZ4     IF NOT X REGISTER BRANCH                  DBG=IDP    3467    A
DBG=IDP            SA1    DAZ=I                                                 DBG=IDP    3468    A
DBG=IDP            SA2    DAZ=XJP+X1-1R0    (X2) = X REG BRANCH SKELETON        DBG=IDP    3469    A
DBG=IDP                                                                         DBG=IDP    3470    A
DBG=IDP  *         GENERATE ...OPI....      (.=BLANK(55B))                       DBG=IDP    3471    A
DBG=IDP                                                                         DBG=IDP    3472    A
DBG=IDP   DAZ4     SA1    DAZ=OPI     (X1) = ...000....   (0=00B)               DBG=IDP    3473    A
DBG=IDP            MX0    -2*CHAR                                               DBG=IDP    3474    A
DBG=IDP            BX6    -X0*X2      (X6) = 2R<OP>                             DBG=IDP    3475    A
DBG=IDP            LX6    7*CHAR-2*CHAR                                         DBG=IDP    3476    A
DBG=IDP            MX0    -4                                                    DBG=IDP    3477    A
DBG=IDP            LX2    3+4         (X2) = 56/OTHER STUFF,4/1ST DESCRIPTOR    DBG=IDP    3478    A
DBG=IDP            BX3    -X0*X2                                                DBG=IDP    3479    A
DBG=IDP            SA4    DAZ=I-1+X3                                            DBG=IDP    3480    A
DBG=IDP            BX6    X1+X6       (X6) = ...OP0....   (0=00B)               DBG=IDP    3481    A
DBG=IDP            LX4    5*CHAR-1*CHAR                                         DBG=IDP    3482    A
DBG=IDP            BX6    X6+X4       (X6) = ...OPI....                         DBG=IDP    3483    A
DBG=IDP                                                                         DBG=IDP    3484    A
DBG=IDP  *         ASSEMBLE NORMAL (I,J,K) ITEMS.                               DBG=IDP    3485    A
DBG=IDP                                                                         DBG=IDP    3486    A
DBG=IDP   DAZ5     SA6    A6+B1                                                 DBG=IDP    3487    A
DBG=IDP            SB6    10*CHAR                                               DBG=IDP    3488    A
DBG=IDP            MX6    0                                                     DBG=IDP    3489    A
DBG=IDP                                                                         DBG=IDP    3490    A
DBG=IDP   DAZ6     LX2    4                                                     DBG=IDP    3491    A
DBG=IDP            SB6    B6-CHAR                                               DBG=IDP    3492    A
DBG=IDP            BX3    -X0*X2      (X3) = DESCRIPTOR                         DBG=IDP    3493    A
DBG=IDP            SA4    DAZ=I-1+X3                                            DBG=IDP    3494    A
DBG=IDP            ZR     X3,DAZ11    IF END OF DESCRIPTOR LIST IN SKELETON     DBG=IDP    3495    A
DBG=IDP                                                                         DBG=IDP    3496    A
DBG=IDP   DAZ7     MI     X4,DAZ8     IF THIS IS A CONDITIONAL ITEM             DBG=IDP    3497    A
DBG=IDP            LX4    X4,B6                                                 DBG=IDP    3498    A
DBG=IDP            BX6    X6+X4                                                 DBG=IDP    3499    A
DBG=IDP            GT     B6,B0,DAZ6  IF PACKING REG (X6) NOT FULL              DBG=IDP    3500    A
DBG=IDP            EQ     DAZ5                                                  DBG=IDP    3501    A
DBG=IDP                                                                         DBG=IDP    3502    A
DBG=IDP  *         PROCESS AND ASSEMBLE CONDITONAL ITEM.                        DBG=IDP    3503    A
DBG=IDP                                                                         DBG=IDP    3504    A
DBG=IDP   DAZ8     NZ     X4,DAZ9     IF *Q* 18 BIT ADDRESS                     DBG=IDP    3505    A
DBG=IDP                                                                         DBG=IDP    3506    A
DBG=IDP  *         HERE TO PROCESS CONDITIONAL *B0* ASSEMBLY.                   DBG=IDP    3507    A
DBG=IDP                                                                         DBG=IDP    3508    A
DBG=IDP            LX2    4                                                     DBG=IDP    3509    A
DBG=IDP            BX3    -X0*X2                                                DBG=IDP    3510    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            SA4    DAZ=I-1+X3                                          DBG=IDP    3511    A
DBG=IDP            SX7    X4-1R0                                              DBG=IDP    3512    A
DBG=IDP            SX4    1RB                                                 DBG=IDP    3513    A
DBG=IDP            LX2    -4          RESTORE (X2)                            DBG=IDP    3514    A
DBG=IDP            NZ     X7,DAZ7     IF NOT *B0* REFERENCE                   DBG=IDP    3515    A
DBG=IDP                                                                       DBG=IDP    3516    A
DBG=IDP  *         HERE TO IGNORE *B0* REFERENCE.                            DBG=IDP    3517    A
DBG=IDP                                                                       DBG=IDP    3518    A
DBG=IDP            LX2    2*4         SKIP OVER *B0* AND SEPARATOR            DBG=IDP    3519    A
DBG=IDP            SB6    B6+CHAR                                             DBG=IDP    3520    A
DBG=IDP            EQ     DAZ6                                                DBG=IDP    3521    A
DBG=IDP                                                                       DBG=IDP    3522    A
DBG=IDP  *         HERE TO PROCESS *Q* 18 BIT ADDRESS.                       DBG=IDP    3523    A
DBG=IDP                                                                       DBG=IDP    3524    A
DBG=IDP   DAZ9     SA1    DAZ=ADR+1   (X1) = 0L<Q>   (ALWAYS .LE. 6 CHARS)   DBG=IDP    3525    A
DBG=IDP            MX0    -CHAR                                               DBG=IDP    3526    A
DBG=IDP                                                                       DBG=IDP    3527    A
DBG=IDP  *         ASSEMBLE *Q*.                                             DBG=IDP    3528    A
DBG=IDP                                                                       DBG=IDP    3529    A
DBG=IDP   DAZ10    LX1    CHAR                                                DBG=IDP    3530    A
DBG=IDP            BX4    -X0*X1                                              DBG=IDP    3531    A
DBG=IDP            ZR     X4,DAZ11    IF FINISHED ASSEMBLING *Q*              DBG=IDP    3532    A
DBG=IDP            LX4    X4,B6                                               DBG=IDP    3533    A
DBG=IDP            SB6    B6-CHAR                                             DBG=IDP    3534    A
DBG=IDP            BX6    X6+X4                                               DBG=IDP    3535    A
DBG=IDP            GE     B6,B0,DAZ10 IF PACKING REG (X6) NOT FULL            DBG=IDP    3536    A
DBG=IDP            SA6    A6+B1                                               DBG=IDP    3537    A
DBG=IDP            MX6    0                                                   DBG=IDP    3538    A
DBG=IDP            SB6    10*CHAR-1*CHAR                                      DBG=IDP    3539    A
DBG=IDP            EQ     DAZ10                                               DBG=IDP    3540    A
DBG=IDP                                                                       DBG=IDP    3541    A
DBG=IDP  *         STORE FINAL WORD.                                         DBG=IDP    3542    A
DBG=IDP                                                                       DBG=IDP    3543    A
DBG=IDP   DAZ11    BX1    X6                                                  DBG=IDP    3544    A
DBG=IDP            RJ     =XSFN       SPACE FILL NAME                         DBG=IDP    3545    A
DBG=IDP            SA6    A6+B1                                               DBG=IDP    3546    A
DBG=IDP                                                                       DBG=IDP    3547    A
DBG=IDP  *         GENERATE RELATIVE ADDRESS.                                DBG=IDP    3548    A
DBG=IDP                                                                       DBG=IDP    3549    A
DBG=IDP   #FRA     IF     DEF,FRA=                                            DBG=IDP    3550    A
DBG=IDP            SA1    DAZ=ADR                                             DBG=IDP    3551    A
DBG=IDP            SA2    =10H                                                DBG=IDP    3552    A
DBG=IDP            SB3    SNAPLNE+5                                           DBG=IDP    3553    A
DBG=IDP            SB2    A6                                                  DBG=IDP    3554    A
DBG=IDP            SB7    DAZ11B      (B7) = RETURN ADDRESS FOR *FRA=*        DBG=IDP    3555    A
DBG=IDP            SA0    B4          SAVE (B4)                               DBG=IDP    3556    A
DBG=IDP            GE     B2,B3,DAZ11A IF WE DONT NEED BLANK FILL             DBG=IDP    3557    A
DBG=IDP            BX6    X2                                                  DBG=IDP    3558    A
DBG=IDP            SA6    A6+B1                                               DBG=IDP    3559    A
DBG=IDP                                                                       DBG=IDP    3560    A
DBG=IDP   DAZ11A   SB6    30                                                  DBG=IDP    3561    A
DBG=IDP            MI     X1,DAZ11C   IF ADDR IS .MI. (I.E. NOT AN ADDRESS)   DBG=IDP    3562    A
```

```
         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            LT     B5,B6,DAZ11C IF 15 BIT INSTRUCTION, NO RELATIVE ADDR          DBG=IDP    3563    A
DBG=IDP            EQ     =XFRA=      FIND RELATIVE ADDRESS                             DBG=IDP    3564    A
DBG=IDP                                                                                 DBG=IDP    3565    A
DBG=IDP   DAZ11B   SA1    A6          (X1) = XX........   .=BLANK(55B)                  DBG=IDP    3566    A
DBG=IDP  *                              X=DIFFERENT CHARS DEPENDING ON DEASSEMBLY       DBG=IDP    3567    A
DBG=IDP            SB4    A0          RESTORE (B4)                                      DBG=IDP    3568    A
DBG=IDP            MX0    3*CHAR                                                        DBG=IDP    3569    A
DBG=IDP            BX1    X0*X1       (X1) = XX.0000000   (.=BLANK(55B),0=00B)          DBG=IDP    3570    A
DBG=IDP            LX6    -3*CHAR                                                       DBG=IDP    3571    A
DBG=IDP            BX6    -X0*X6      (X6) = 000NNNNNN.   (.=BLANK(55B),0=00B)          DBG=IDP    3572    A
DBG=IDP            BX6    X6+X1       (X6) = XX.NNNNNN.   (.=BLANK(55B))                DBG=IDP    3573    A
DBG=IDP            SA6    A1                                                            DBG=IDP    3574    A
DBG=IDP            SX1    3RIN                                                          DBG=IDP    3575    A
DBG=IDP            BX6    X7+X1       (X6) = XXXXXXXIN.   (.=BLANK(55B))                DBG=IDP    3576    A
DBG=IDP            LX6    10*CHAR-3*CHAR                                                DBG=IDP    3577    A
DBG=IDP            SA6    A6+B1                                                         DBG=IDP    3578    A
DBG=IDP   #FRA     ENDIF                                                                DBG=IDP    3579    A
DBG=IDP                                                                                 DBG=IDP    3580    A
DBG=IDP  *         FINAL PROCESSING AND EXIT.                                           DBG=IDP    3581    A
DBG=IDP                                                                                 DBG=IDP    3582    A
DBG=IDP   DAZ11C   MX6    0                                                             DBG=IDP    3583    A
DBG=IDP            SA6    A6+B1                                                         DBG=IDP    3584    A
DBG=IDP            SX0    B5                                                            DBG=IDP    3585    A
DBG=IDP            PRIDP  =XSNAPLNE                                                     DBG=IDP    3586    A
DBG=IDP            SB5    X0                                                            DBG=IDP    3587    A
DBG=IDP            SB6    59-57                                                         DBG=IDP    3588    A
DBG=IDP            SA2    DAZ=OP                                                        DBG=IDP    3589    A
DBG=IDP            SA2    DAZ=PS+X2                                                     DBG=IDP    3590    A
DBG=IDP            LX1    X2,B6       SET TO *FORCED UPPER(.MI.)/NOT UPPER(.PL.)*       DBG=IDP    3591    A
DBG=IDP            EQ     EXIT.                                                         DBG=IDP    3592    A
DBG=IDP                                                                                 DBG=IDP    3593    A
DBG=IDP  *         HERE IF OUTPUTING REST OF WORD AS DATA.                              DBG=IDP    3594    A
DBG=IDP                                                                                 DBG=IDP    3595    A
DBG=IDP   DAZ12    SB5    B4          (B5) = LENGTH OF REMAINDER OF INSTR WORD          DBG=IDP    3596    A
DBG=IDP            SA1    A5          (X1) = INSTRUCTION WORD                           DBG=IDP    3597    A
DBG=IDP            RJ     VFD         VARIABLE FIELD DEFINITION                         DBG=IDP    3598    A
DBG=IDP            SA6    A6+B1                                                         DBG=IDP    3599    A
DBG=IDP            SA7    A6+B1                                                         DBG=IDP    3600    A
DBG=IDP            SA1    =7L   DATA                                                    DBG=IDP    3601    A
DBG=IDP            BX6    X1                                                            DBG=IDP    3602    A
DBG=IDP            SA6    A7+B1                                                         DBG=IDP    3603    A
DBG=IDP            SX0    B5          SAVE (B5)                                         DBG=IDP    3604    A
DBG=IDP            PRIDP  =XSNAPLNE                                                     DBG=IDP    3605    A
DBG=IDP            SB5    X0          RESTORE (B5)                                      DBG=IDP    3606    A
DBG=IDP            MX2    0           SET TO *THIS IS A DATA ITEM*                      DBG=IDP    3607    A
DBG=IDP            MX1    1           SET TO *FORCED UPPER*                             DBG=IDP    3608    A
DBG=IDP            EQ     EXIT.                                                         DBG=IDP    3609    A
DBG=IDP                                                                                 DBG=IDP    3610    A
DBG=IDP                                                                                 DBG=IDP    3611    A
DBG=IDP   DAZ=ADR  BSSZ   1           18 BIT *Q* IN BINARY                              DBG=IDP    3612    A
DBG=IDP            BSSZ   1           18 BIT *Q* IN -0L- FORMAT                         DBG=IDP    3613    A
DBG=IDP   DAZ=OPI  CON    3L   +4R                                                     DBG=IDP    3614    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP   DAZ=OP   DATA   0          OP CODE FOR INSTRUCTION                          DBG=IDP    3615    A
DBG=IDP   DAZ=I    DATA   0          *I* REGISTER PORTION OF INSTRUCTION              DBG=IDP    3616    A
DBG=IDP   DAZ=J    DATA   0          *J* REGISTER PORTION OF INSTRUCTION              DBG=IDP    3617    A
DBG=IDP   DAZ=K    DATA   0          *K* REGISTER PORTION OF INSTRUCTION              DBG=IDP    3618    A
DBG=IDP   DAZ=Q    DATA   -1         FLAG TO INDICATE 18 BIT *Q*                      DBG=IDP    3619    A
DBG=IDP            DATA   1R+                                                         DBG=IDP    3620    A
DBG=IDP            DATA   1R-                                                         DBG=IDP    3621    A
DBG=IDP            DATA   1R*                                                         DBG=IDP    3622    A
DBG=IDP            DATA   1R/                                                         DBG=IDP    3623    A
DBG=IDP            DATA   1R,                                                         DBG=IDP    3624    A
DBG=IDP            DATA   1RA                                                         DBG=IDP    3625    A
DBG=IDP            DATA   1RB                                                         DBG=IDP    3626    A
DBG=IDP            DATA   -0         FLAG TO INDICATE CONDITIONAL *B0* ASSEMBLY       DBG=IDP    3627    A
DBG=IDP            DATA   1RX                                                         DBG=IDP    3628    A
DBG=IDP            DATA   1R                                                          DBG=IDP    3629    A
DBG=IDP   INST     SPACE  4,8                                                         DBG=IDP    3630    A
DBG=IDP  **        INST - MACRO TO GENERATE INSTRUCTION SKELETONS.                    DBG=IDP    3631    A
DBG=IDP  *         GENERATES TABLE OF INSTRUCTION SKELETONS USED BY *PIG* TO          DBG=IDP    3632    A
DBG=IDP  *                CONVERT INSTRUCTIONS TO HUMAN-READABLE FORM FOR THE         DBG=IDP    3633    A
DBG=IDP  *                OBJECT CODE LISTING.                                        DBG=IDP    3634    A
DBG=IDP  *                                                                            DBG=IDP    3635    A
DBG=IDP  *         INST   (KEY),BJMP,FORCE,IJJ                                        DBG=IDP    3636    A
DBG=IDP  *                                                                            DBG=IDP    3637    A
DBG=IDP  *         *KEY*  = INSTRUCTION DESCRIPTOR.  EACH CHARACTER OF THE *KEY*      DBG=IDP    3638    A
DBG=IDP  *                  HAS MEANING AS FOLLOWS --                                 DBG=IDP    3639    A
DBG=IDP  *                           I   I-PORTION OF INSTRUCTION                     DBG=IDP    3640    A
DBG=IDP  *                           J   J-PORTION OF INSTRUCTION                     DBG=IDP    3641    A
DBG=IDP  *                           K   K-PORTION OF INSTRUCTION (3 BITS ONLY)       DBG=IDP    3642    A
DBG=IDP  *                           Q   18-BIT *K* ADDRESS FIELD (MUST BE LAST)      DBG=IDP    3643    A
DBG=IDP  *                           C   OUTPUT A *B* IF THE NEXT REGISTER IS         DBG=IDP    3644    A
DBG=IDP  *                                       NOT A ZERO,                          DBG=IDP    3645    A
DBG=IDP  *                                       ELSE, SKIP THE NEXT 2 ITEMS          DBG=IDP    3646    A
DBG=IDP  *                           + - * / A B X , AND BLANK ALL STAND FOR          DBG=IDP    3647    A
DBG=IDP  *                                       THEMSELVES.                          DBG=IDP    3648    A
DBG=IDP  *         *BJMP* = NON-EMPTY IF THIS A *B-REGISTER* JUMP INSTRUCTION.        DBG=IDP    3649    A
DBG=IDP  *                           (04 THRU 07 INSTRUCTION.)                        DBG=IDP    3650    A
DBG=IDP  *         *FORCE*= THIS INSTRUCTION FORCES THE NEXT INSTRUCTION UPPER.       DBG=IDP    3651    A
DBG=IDP  *         *COPY* = COPY *J* INTO *K*.                                        DBG=IDP    3652    A
DBG=IDP                                                                               DBG=IDP    3653    A
DBG=IDP   P.COPY   =      2*CHAR                                                      DBG=IDP    3654    A
DBG=IDP            NOREF  D,L                                                         DBG=IDP    3655    A
DBG=IDP                                                                               DBG=IDP    3656    A
DBG=IDP                                                                               DBG=IDP    3657    A
DBG=IDP   INST     MACRO  KEY,BJMP,FORCE,IJJ                                          DBG=IDP    3658    A
DBG=IDP   A        MICRO  3,1,=KEY=                                                   DBG=IDP    3659    A
DBG=IDP   B        MICRO  5,,=KEY=                                                    DBG=IDP    3660    A
DBG=IDP   B        MICRO  1,,="A""B"=                                                 DBG=IDP    3661    A
DBG=IDP   C        MICCNT B                                                           DBG=IDP    3662    A
DBG=IDP   D        SET    0                                                           DBG=IDP    3663    A
DBG=IDP   L        SET    0                                                           DBG=IDP    3664    A
DBG=IDP   A        MICRO  C+3,1,=KEY=                                                 DBG=IDP    3665    A
DBG=IDP            IFC    EQ,="A"=Q=,1                                                DBG=IDP    3666    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP   L        SET    L+4                                                  DBG=IDP    3667    A
DBG=IDP            IFC    NE,,BJMP,,,1                                         DBG=IDP    3668    A
DBG=IDP   L        SET    L+2                                                  DBG=IDP    3669    A
DBG=IDP            IFC    NE,,FORCE,,,1                                        DBG=IDP    3670    A
DBG=IDP   L        SET    L+1                                                  DBG=IDP    3671    A
DBG=IDP            VFD    3/L                                                  DBG=IDP    3672    A
DBG=IDP  .1        DUP    C                                                    DBG=IDP    3673    A
DBG=IDP   D        SET    D+1                                                  DBG=IDP    3674    A
DBG=IDP   A        MICRO  D,1,="B"=                                            DBG=IDP    3675    A
DBG=IDP   A        MICRO  2*1R"A"-1,2,/101112-D-E-F-G-H010203-L-M-N-O-P04-R-S-T-U  DBG=IDP    3676    A
DBG=IDP  ,-V-W13-Y-Z-0-1-2-3-4-5-6-7-8-905060708-(-)-$-=1409/                  DBG=IDP    3677    A
DBG=IDP            VFD    4/"A"                                                DBG=IDP    3678    A
DBG=IDP  .1        ENDD                                                        DBG=IDP    3679    A
DBG=IDP            IFLE   $,2*6-1+4,1                                          DBG=IDP    3680    A
DBG=IDP   8        ERR    INST - DESCRIPTOR (KEY) IS TOO LONG                  DBG=IDP    3681    A
DBG=IDP            POS    P.COPY+1                                             DBG=IDP    3682    A
DBG=IDP   A        MICRO  1,2, KEY                                             DBG=IDP    3683    A
DBG=IDP            VFD    1/IJJ,12/2R"A"                                       DBG=IDP    3684    A
DBG=IDP   INST     ENDM                                                        DBG=IDP    3685    A
DBG=IDP   DAZ=PS   SPACE  4,8                                                  DBG=IDP    3686    A
DBG=IDP  **        DAZ=PS - INSTRUCTION SKELETON TABLES FOR *DAZ*.             DBG=IDP    3687    A
DBG=IDP                                                                        DBG=IDP    3688    A
DBG=IDP                                                                        DBG=IDP    3689    A
DBG=IDP   DAZ=PS   INST   (PS  Q)              00    (FAKE)                    DBG=IDP    3690    A
DBG=IDP            INST   (RJ  Q),,1           01                              DBG=IDP    3691    A
DBG=IDP            INST   (JP  CI+Q),,1        02                              DBG=IDP    3692    A
DBG=IDP            INST   (JXI XJ,Q)           03I   (FAKE)                    DBG=IDP    3693    A
DBG=IDP   DAZ=EQ   INST   (EQ  CI,CJ,Q),1      04                              DBG=IDP    3694    A
DBG=IDP            INST   (NE  CI,CJ,Q),1      05                              DBG=IDP    3695    A
DBG=IDP            INST   (GE  BI,CJ,Q),1      06                              DBG=IDP    3696    A
DBG=IDP            INST   (LT  BI,CJ,Q),1      07                              DBG=IDP    3697    A
DBG=IDP                                                                        DBG=IDP    3698    A
DBG=IDP            INST   (BXI XJ),,,1         10                              DBG=IDP    3699    A
DBG=IDP            INST   (BXI XJ*XK)          11                              DBG=IDP    3700    A
DBG=IDP            INST   (BXI XJ+XK)          12                              DBG=IDP    3701    A
DBG=IDP            INST   (BXI XJ-XK)          13                              DBG=IDP    3702    A
DBG=IDP            INST   (BXI -XJ),,,1        14                              DBG=IDP    3703    A
DBG=IDP            INST   (BXI -XK*XJ)         15                              DBG=IDP    3704    A
DBG=IDP            INST   (BXI -XK+XJ)         16                              DBG=IDP    3705    A
DBG=IDP            INST   (BXI -XJ-XK)         17                              DBG=IDP    3706    A
DBG=IDP                                                                        DBG=IDP    3707    A
DBG=IDP            INST   (LXI JKB)            20                              DBG=IDP    3708    A
DBG=IDP            INST   (AXI JKB)            21                              DBG=IDP    3709    A
DBG=IDP            INST   (LXI CJ,XK)          22                              DBG=IDP    3710    A
DBG=IDP            INST   (AXI CJ,XK)          23                              DBG=IDP    3711    A
DBG=IDP            INST   (NXI CJ,XK)          24                              DBG=IDP    3712    A
DBG=IDP            INST   (ZXI CJ,XK)          25                              DBG=IDP    3713    A
DBG=IDP            INST   (UXI CJ,XK)          26                              DBG=IDP    3714    A
DBG=IDP            INST   (PXI CJ,XK)          27                              DBG=IDP    3715    A
DBG=IDP                                                                        DBG=IDP    3716    A
DBG=IDP            INST   (FXI XJ+XK)          30                              DBG=IDP    3717    A
DBG=IDP            INST   (FXI XJ-XK)          31                              DBG=IDP    3718    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            INST   (DXI XJ+XK)          32                                  DBG=IDP     3719     A
DBG=IDP            INST   (DXI XJ-XK)          33                                  DBG=IDP     3720     A
DBG=IDP            INST   (RXI XJ+XK)          34                                  DBG=IDP     3721     A
DBG=IDP            INST   (RXI XJ-XK)          35                                  DBG=IDP     3722     A
DBG=IDP            INST   (IXI XJ+XK)          36                                  DBG=IDP     3723     A
DBG=IDP            INST   (IXI XJ-XK)          37                                  DBG=IDP     3724     A
DBG=IDP                                                                            DBG=IDP     3725     A
DBG=IDP            INST   (FXI XJ*XK)          40                                  DBG=IDP     3726     A
DBG=IDP            INST   (RXI XJ*XK)          41                                  DBG=IDP     3727     A
DBG=IDP            INST   (DXI XJ*XK)          42                                  DBG=IDP     3728     A
DBG=IDP            INST   (MXI JKB)            43                                  DBG=IDP     3729     A
DBG=IDP            INST   (FXI XJ/XK)          44                                  DBG=IDP     3730     A
DBG=IDP            INST   (RXI XJ/XK)          45                                  DBG=IDP     3731     A
DBG=IDP            INST   (NO  IJKB)           46                                  DBG=IDP     3732     A
DBG=IDP            INST   (CXI XK),,,1         47                                  DBG=IDP     3733     A
DBG=IDP                                                                            DBG=IDP     3734     A
DBG=IDP            INST   (SAI AJ+Q)           50                                  DBG=IDP     3735     A
DBG=IDP            INST   (SAI CJ+Q)           51                                  DBG=IDP     3736     A
DBG=IDP            INST   (SAI XJ+Q)           52                                  DBG=IDP     3737     A
DBG=IDP            INST   (SAI CK+XJ)          53                                  DBG=IDP     3738     A
DBG=IDP            INST   (SAI CK+AJ)          54                                  DBG=IDP     3739     A
DBG=IDP            INST   (SAI AJ-BK)          55                                  DBG=IDP     3740     A
DBG=IDP            INST   (SAI CK+BJ)          56                                  DBG=IDP     3741     A
DBG=IDP            INST   (SAI -BK+BJ)         57                                  DBG=IDP     3742     A
DBG=IDP                                                                            DBG=IDP     3743     A
DBG=IDP            INST   (SBI AJ+Q)           60                                  DBG=IDP     3744     A
DBG=IDP            INST   (SBI CJ+Q)           61                                  DBG=IDP     3745     A
DBG=IDP            INST   (SBI XJ+Q)           62                                  DBG=IDP     3746     A
DBG=IDP            INST   (SBI CK+XJ)          63                                  DBG=IDP     3747     A
DBG=IDP            INST   (SBI CK+AJ)          64                                  DBG=IDP     3748     A
DBG=IDP            INST   (SBI AJ-BK)          65                                  DBG=IDP     3749     A
DBG=IDP            INST   (SBI CK+BJ)          66                                  DBG=IDP     3750     A
DBG=IDP            INST   (SBI -BK+BJ)         67                                  DBG=IDP     3751     A
DBG=IDP                                                                            DBG=IDP     3752     A
DBG=IDP            INST   (SXI AJ+Q)           70                                  DBG=IDP     3753     A
DBG=IDP            INST   (SXI CJ+Q)           71                                  DBG=IDP     3754     A
DBG=IDP            INST   (SXI XJ+Q)           72                                  DBG=IDP     3755     A
DBG=IDP            INST   (SXI CK+XJ)          73                                  DBG=IDP     3756     A
DBG=IDP            INST   (SXI CK+AJ)          74                                  DBG=IDP     3757     A
DBG=IDP            INST   (SXI AJ-BK)          75                                  DBG=IDP     3758     A
DBG=IDP            INST   (SXI CK+BJ)          76                                  DBG=IDP     3759     A
DBG=IDP            INST   (SXI -BK+BJ)         77                                  DBG=IDP     3760     A
DBG=IDP                                                                            DBG=IDP     3761     A
DBG=IDP   DAZ=XJP  INST   (ZR  XJ,Q)           030                                 DBG=IDP     3762     A
DBG=IDP            INST   (NZ  XJ,Q)           031                                 DBG=IDP     3763     A
DBG=IDP            INST   (PL  XJ,Q)           032                                 DBG=IDP     3764     A
DBG=IDP            INST   (MI  XJ,Q)           033                                 DBG=IDP     3765     A
DBG=IDP            INST   (IR  XJ,Q)           034                                 DBG=IDP     3766     A
DBG=IDP            INST   (OR  XJ,Q)           035                                 DBG=IDP     3767     A
DBG=IDP            INST   (DF  XJ,Q)           036                                 DBG=IDP     3768     A
DBG=IDP            INST   (ID  XJ,Q)           037                                 DBG=IDP     3769     A
DBG=IDP   DOD      SPACE  4,8                                                      DBG=IDP     3770     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  **        DOD - DUMP CENTRAL MEMORY -- OCTAL AND DPC.                          DBG=IDP    3771    A
DBG=IDP  *                                                                              DBG=IDP    3772    A
DBG=IDP  *                                                                              DBG=IDP    3773    A
DBG=IDP  *                CONVERTS CM WORDS TO OCTAL DPC AND ALSO DUMPS THE             DBG=IDP    3774    A
DBG=IDP  *         UNCONVERTED DPC EQUIVALENT IN THE FOLLOWING FORMAT--                 DBG=IDP    3775    A
DBG=IDP  *     COL 1         1         1         1         1         1                  DBG=IDP    3776    A
DBG=IDP  *         .NNNNNN.+.CCCCCCCCCCCCCCCCCCCC.........UUUUUUUUUU          001000     DBG=IDP    3777    A
DBG=IDP  *         N = ADDR OF WORD DUMPED                                              DBG=IDP    3778    A
DBG=IDP  *         + = + IF *NNNNNN* IS DECK RELATIVE, ELSE                  001020     DBG=IDP    3779    A
DBG=IDP  *           = BLANK(55B) IF *NNNNNN* IS ABSOLUTE ADDR.              001030     DBG=IDP    3780    A
DBG=IDP  *         C = CONVERTED CM WORD                                                DBG=IDP    3781    A
DBG=IDP  *         U = UNCONVERTED CM WORD. NOTE--TRAILING COLONS ARE TREATED AS        DBG=IDP    3782    A
DBG=IDP  *               AN EOL AND THEREFORE WILL NOT APPEAR.                          DBG=IDP    3783    A
DBG=IDP  *         . = BLANK(55B)                                                       DBG=IDP    3784    A
DBG=IDP  *                                                                              DBG=IDP    3785    A
DBG=IDP  *         ENTRY  (X2) = 18/0,21/LEV OF IND ADDR,21/FWA                         DBG=IDP    3786    A
DBG=IDP  *                (X3) = 18/0,21/LEV OF IND ADDR,21/LWA                         DBG=IDP    3787    A
DBG=IDP  *                       NOTE-- (X3) IS OPTIONAL                                DBG=IDP    3788    A
DBG=IDP  *                (X4) = 18/0,21/LEV OF IND ADDR,21/LEN                         DBG=IDP    3789    A
DBG=IDP  *                         NOTE-- IF LEN = 0, DMP IS FROM FWA TO LWA,           DBG=IDP    3790    A
DBG=IDP  *                         ELSE DMP IS FROM FWA TO FWA+LEN-1                    DBG=IDP    3791    A
DBG=IDP  *                                                                              DBG=IDP    3792    A
DBG=IDP  *         EXIT   (B2) = .MI. IF FWA,LWA, OR LEN IS BAD, ELSE .PL.              DBG=IDP    3793    A
DBG=IDP  *                                                                              DBG=IDP    3794    A
DBG=IDP  *         USES   ALL                        (INCLUDES ALL CALLS)    001050     DBG=IDP    3795    A
DBG=IDP  *                                                                              DBG=IDP    3796    A
DBG=IDP  *         CALLS  CAD,FLL,PRIDP,WOD                                  001070     DBG=IDP    3797    A
DBG=IDP                                                                                 DBG=IDP    3798    A
DBG=IDP                                                                                 DBG=IDP    3799    A
DBG=IDP   DOD      SUBR                ** ENTRY/EXIT **                                 DBG=IDP    3800    A
DBG=IDP            RJ     FLL          CHECK FWA,LWA, AND LENGTH                        DBG=IDP    3801    A
DBG=IDP            MI     B2,EXIT.     IF FWA,LWA, OR LEN IS BAD                        DBG=IDP    3802    A
DBG=IDP                                                                                 DBG=IDP    3803    A
DBG=IDP  *         HERE TO BUILD SNAP OUTPUT LINE.                                      DBG=IDP    3804    A
DBG=IDP                                                                                 DBG=IDP    3805    A
DBG=IDP   DOD2     SA1    AP=FWA       (X1) = ADDR OF NEXT WORD TO DUMP                 DBG=IDP    3806    A
DBG=IDP            SA2    A1+B1        (X2) = LWA TO DUMP                               DBG=IDP    3807    A
DBG=IDP            IX3    X2-X1                                                         DBG=IDP    3808    A
DBG=IDP            SX6    X1+B1                                                         DBG=IDP    3809    A
DBG=IDP            SB2    B0           SET TO *NO ERROR OCCURRED*                       DBG=IDP    3810    A
DBG=IDP            MI     X3,EXIT.     IF FINISHED                                      DBG=IDP    3811    A
DBG=IDP            SA6    A1                                                            DBG=IDP    3812    A
DBG=IDP            SA0    X1                                                            DBG=IDP    3813    A
DBG=IDP            RJ     CAD          CONVERT ADDR TO DPC                      001090  DBG=IDP    3814    A
DBG=IDP            SA6    SNAPLNE                                                       DBG=IDP    3815    A
DBG=IDP            SA1    A0           (X1) = NEXT WORD TO BE DUMPED                    DBG=IDP    3816    A
DBG=IDP            RJ     =XWOD        CONVERT A FULL BINARY WORD TO OCTAL DPC          DBG=IDP    3817    A
DBG=IDP            SA6    A6+B1                                                         DBG=IDP    3818    A
DBG=IDP            SA7    A6+B1                                                         DBG=IDP    3819    A
DBG=IDP            SA1    =10H                                                          DBG=IDP    3820    A
DBG=IDP            SA2    A0                                                            DBG=IDP    3821    A
DBG=IDP            BX6    X1                                                            DBG=IDP    3822    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            LX7    X2                                                    DBG=IDP     3823     A
DBG=IDP            SA6    A7+B1                                                 DBG=IDP     3824     A
DBG=IDP            SA7    A6+B1                                                 DBG=IDP     3825     A
DBG=IDP            MX6    0                                                     DBG=IDP     3826     A
DBG=IDP            SA6    A7+B1       MARK EOL                                  DBG=IDP     3827     A
DBG=IDP            PRIDP  SNAPLNE                                               DBG=IDP     3828     A
DBG=IDP            EQ     DOD2                                                  DBG=IDP     3829     A
DBG=IDP   FRZ      SPACE  4,8                                          000710   DBG=IDP     3830     A
DBG=IDP  **        FRZ - FREEZE INTERACTIVE SESSION.                   000720   DBG=IDP     3831     A
DBG=IDP  *                                                             000730   DBG=IDP     3832     A
DBG=IDP  *                                                             000740   DBG=IDP     3833     A
DBG=IDP  *         PERFORMS A CHECKPOINT OF AN INTERACTIVE SESSION BY  000750   DBG=IDP     3834     A
DBG=IDP  *         WRITING THE CURRENT CORE IMAGE TO LFN *F.FRZ* IN A  000760   DBG=IDP     3835     A
DBG=IDP  *         SPECIAL FORMAT.                                     000770   DBG=IDP     3836     A
DBG=IDP  *                                                             000780   DBG=IDP     3837     A
DBG=IDP  *         *F.FRZ* FORMAT --                                   000790   DBG=IDP     3838     A
DBG=IDP  *                                                             000800   DBG=IDP     3839     A
DBG=IDP  *           RECORD 1 CONSISTS OF A LOADER ABSOLUTE BINARY, CALLED *IFR* 000810   DBG=IDP     3840     A
DBG=IDP  *           (IDP FREEZE RESTART), THAT WHEN EXECUTED WILL READ THE      000820   DBG=IDP     3841     A
DBG=IDP  *           FROZEN HOST BACK INTO CORE (OVERLAYING ITSELF), AND         000830   DBG=IDP     3842     A
DBG=IDP  *           THEN REINVOKE *IDP*.                                        000840   DBG=IDP     3843     A
DBG=IDP  *                                                                       000850   DBG=IDP     3844     A
DBG=IDP  *           RECORD 2 CONTAINS THE CORE IMAGE OF THE FROZEN HOST.        000860   DBG=IDP     3845     A
DBG=IDP  *                                                                       000870   DBG=IDP     3846     A
DBG=IDP  *         SEE ROUTINE *IFR* IN DATA SECTION BEGINNING AT ADDR *FW.IFR*. 000880   DBG=IDP     3847     A
DBG=IDP  *                                                             000890   DBG=IDP     3848     A
DBG=IDP  *         ENTRY  NONE                                         000900   DBG=IDP     3849     A
DBG=IDP  *                                                             000910   DBG=IDP     3850     A
DBG=IDP  *         EXIT   *F.FRZ* WRITTEN                              000920   DBG=IDP     3851     A
DBG=IDP  *                                                             000930   DBG=IDP     3852     A
DBG=IDP  *         USES   ALL BUT A0,X0,A5,X5   (INCLUDES ALL CALLS)   000940   DBG=IDP     3853     A
DBG=IDP  *                                                             000950   DBG=IDP     3854     A
DBG=IDP  *         CALLS  MEMORY,REWIND,WRITER,WRITEW                  000960   DBG=IDP     3855     A
DBG=IDP                                                                000970   DBG=IDP     3856     A
DBG=IDP                                                                000980   DBG=IDP     3857     A
DBG=IDP   FRZ      SUBR               ** ENTRY/EXIT **                 000990   DBG=IDP     3858     A
DBG=IDP                                                                001000   DBG=IDP     3859     A
DBG=IDP  *         FIND OUT ACTUAL FIELD LENGTH SCM (AFLS) SO THAT *IFR*  001010   DBG=IDP     3860     A
DBG=IDP  *         CAN KNOW HOW MUCH IS NEEDED TO RESTART.  IF *CMM* IS ACTIVE,  001020   DBG=IDP     3861     A
DBG=IDP  *         SAVE THIS KNOWLEDGE SO THAT *IFR* CAN INDICATE SO      001030   DBG=IDP     3862     A
DBG=IDP  *         TO OPERATING SYSTEM.                                   001040   DBG=IDP     3863     A
DBG=IDP                                                                   001050   DBG=IDP     3864     A
DBG=IDP            SA1    RA.LWP                                          001060   DBG=IDP     3865     A
DBG=IDP            BX6    X6-X6                                           001070   DBG=IDP     3866     A
DBG=IDP            SX7    X1          (X7) = .MI. IF *CMM* ACTIVE, ELSE .PL.  001080   DBG=IDP     3867     A
DBG=IDP            SA6    MEM#HOST                                        001090   DBG=IDP     3868     A
DBG=IDP            MEMORY SCM,A6,RCL                                      001100   DBG=IDP     3869     A
DBG=IDP            SX2    B1                                              001102   DBG=IDP     3870     A
DBG=IDP            AX7    59          (X7) = -0 IF *CMM* ACTIVE, ELSE +0  001110   DBG=IDP     3871     A
DBG=IDP            SA1    MEM#HOST                                        001120   DBG=IDP     3872     A
DBG=IDP            BX7    X2*X7       (X7) = 1 IF *CMM* ACTIVE, ELSE +0   001122   DBG=IDP     3873     A
DBG=IDP            LX7    2-0                                             001130   DBG=IDP     3874     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            SX7    X7+B1       SET BIT 1S0                                  001140   DBG=IDP     3875     A
DBG=IDP            BX6    X1-X7       SET *CMM* IF APPROPRIATE/CLEAR COMPLETE BIT001150   DBG=IDP     3876     A
DBG=IDP            SA6    A1                                                       001160   DBG=IDP     3877     A
DBG=IDP                                                                            001170   DBG=IDP     3878     A
DBG=IDP  *         NEED TO SAVE 3 SPECIAL WORDS BECAUSE WE ARE ONLY WRITING        001180   DBG=IDP     3879     A
DBG=IDP  *         FROM *RA.MTR+1* TO FL-2 TO *F.FRZ*.                             001190   DBG=IDP     3880     A
DBG=IDP  *           1. SAVE (RA.SSW) BECAUSE APPARENTLY *CIO* WILL SOMETIMES      001200   DBG=IDP     3881     A
DBG=IDP  *              NOT ALLOW ONE TO HAVE *RA.MTR* IN CIO BUFFER AREA.         001210   DBG=IDP     3882     A
DBG=IDP  *           2. SAVE WORDS AT FL-1 AND FL-2 BECAUSE LWA AND LWA+1 OF CIO 001220   DBG=IDP     3883     A
DBG=IDP  *              BUFFER ARE NOT WRITTEN.                                    001230   DBG=IDP     3884     A
DBG=IDP                                                                            001240   DBG=IDP     3885     A
DBG=IDP            AX1    30          (X1) = AFLS OF FROZEN HOST                   001250   DBG=IDP     3886     A
DBG=IDP            SA2    RA.SSW                                                   001260   DBG=IDP     3887     A
DBG=IDP            SA3    X1-2                                                     001270   DBG=IDP     3888     A
DBG=IDP            SA4    A3+B1                                                    001280   DBG=IDP     3889     A
DBG=IDP            BX6    X2                                                       001290   DBG=IDP     3890     A
DBG=IDP            LX7    X3                                                       001300   DBG=IDP     3891     A
DBG=IDP            SA6    FRZ#SAV                                                  001310   DBG=IDP     3892     A
DBG=IDP            SA7    A6+B1                                                    001320   DBG=IDP     3893     A
DBG=IDP            BX6    X4                                                       001330   DBG=IDP     3894     A
DBG=IDP            SA6    A7+B1                                                    001340   DBG=IDP     3895     A
DBG=IDP                                                                            001350   DBG=IDP     3896     A
DBG=IDP  *         SET UP AND *OPEN* *F.FRZ* FET FOR RECORD 1 WRITE.               001360   DBG=IDP     3897     A
DBG=IDP  *         FIRST, THOUGH, WE NEED TO SET THE *COMPLETE* BIT                001370   DBG=IDP     3898     A
DBG=IDP  *         IN *F.FRZ* FET BECAUSE IF WE ARE NOW FREEZING AN                001380   DBG=IDP     3899     A
DBG=IDP  *         INTERACTIVE SESSION THAT WAS PREVIOUSLY FROZEN AND UNFROZEN,    001390   DBG=IDP     3900     A
DBG=IDP  *         THEN *F.FRZ* IS MARKED AS STILL BEING ACTIVE.  THIS HAPPENS     001400   DBG=IDP     3901     A
DBG=IDP  *         BECAUSE WHEN THE PREVIOUS SESSION WAS FROZEN, *F.FRZ* WAS       001410   DBG=IDP     3902     A
DBG=IDP  *         USED TO WRITE ITSELF OUT, AND WAS THEREFORE STILL ACTIVE WHEN 001420   DBG=IDP     3903     A
DBG=IDP  *         CORE IMAGE WAS WRITTEN TO DISK.                                 001430   DBG=IDP     3904     A
DBG=IDP                                                                            001440   DBG=IDP     3905     A
DBG=IDP            SA1    F.FRZ                                                    001450   DBG=IDP     3906     A
DBG=IDP            MX2    -1                                                       001460   DBG=IDP     3907     A
DBG=IDP            BX6    -X2+X1      SET COMPLETE BIT                             001470   DBG=IDP     3908     A
DBG=IDP            SA6    A1                                                       001480   DBG=IDP     3909     A
DBG=IDP                                                                            001490   DBG=IDP     3910     A
DBG=IDP            SX6    FW.IFR                                                   001500   DBG=IDP     3911     A
DBG=IDP            SX7    FW.IFR+L.IFR-1                                           001510   DBG=IDP     3912     A
DBG=IDP            SA6    A6+B1       FIRST  = FW.IFR                              001520   DBG=IDP     3913     A
DBG=IDP            SA7    A6+B1       IN     = FW.IFR+L.IFR-1                      001530   DBG=IDP     3914     A
DBG=IDP            SA6    A7+B1       OUT    = FW.IFR                              001540   DBG=IDP     3915     A
DBG=IDP            SX7    X7+B1                                                    001550   DBG=IDP     3916     A
DBG=IDP            SA7    A6+B1       LIMIT  = FW.IFR+L.IFR                        001560   DBG=IDP     3917     A
DBG=IDP                                                                            001570   DBG=IDP     3918     A
DBG=IDP            OPEN   A1,,RCL                                                  001580   DBG=IDP     3919     A
DBG=IDP            SA1    F.FRZ+4     (X1) = LIMIT                                 001590   DBG=IDP     3920     A
DBG=IDP            SX6    X1-1                                                     001600   DBG=IDP     3921     A
DBG=IDP            SA6    F.FRZ+2     IN = LIMIT-1                                 001610   DBG=IDP     3922     A
DBG=IDP                                                                            001620   DBG=IDP     3923     A
DBG=IDP  *         WRITE INTERACTIVE FREEZE RESTART PROGRAM, *IFR*,                001630   DBG=IDP     3924     A
DBG=IDP  *         TO RECORD 1 OF *F.FRZ*.                                         001640   DBG=IDP     3925     A
DBG=IDP                                                                            001650   DBG=IDP     3926     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            WRITER F.FRZ,,RCL                                    001660   DBG=IDP     3927     A
DBG=IDP                                                                 001670   DBG=IDP     3928     A
DBG=IDP  *         SET UP *F.FRZ* FET SO THAT BUFFER IS RA.MTR+1 THRU FL-1.  001680   DBG=IDP     3929     A
DBG=IDP                                                                 001690   DBG=IDP     3930     A
DBG=IDP            SA1    MEM#HOST                                      001700   DBG=IDP     3931     A
DBG=IDP            SX6    RA.MTR+1     (X6) = FIRST = OUT               001710   DBG=IDP     3932     A
DBG=IDP            LX1    30                                            001720   DBG=IDP     3933     A
DBG=IDP            SX7    X1-2         (X7) = IN = AFLS-2 (FOR SAFETY)  001730   DBG=IDP     3934     A
DBG=IDP            SA6    F.FRZ+1      FIRST = RA.MTR+1                 001740   DBG=IDP     3935     A
DBG=IDP            SA7    A6+B1        IN =AFLS-2                       001750   DBG=IDP     3936     A
DBG=IDP            SA6    A7+B1        OUT = RA.MTR+1                   001760   DBG=IDP     3937     A
DBG=IDP            SX7    X7+B1        (X7) = LIMIT = AFLS-1 (FOR SAFETY)  001770   DBG=IDP     3938     A
DBG=IDP            SA7    A6+B1        LIMIT = AFLS-1                   001780   DBG=IDP     3939     A
DBG=IDP                                                                 001790   DBG=IDP     3940     A
DBG=IDP  *         WRITE ALL OF SCM TO *F.FRZ/RECORD 2*.                001800   DBG=IDP     3941     A
DBG=IDP                                                                 001810   DBG=IDP     3942     A
DBG=IDP            WRITER F.FRZ,,RCL                                    001820   DBG=IDP     3943     A
DBG=IDP            EQ     EXIT.                                         001830   DBG=IDP     3944     A
DBG=IDP                                                                 001840   DBG=IDP     3945     A
DBG=IDP                                                                 001850   DBG=IDP     3946     A
DBG=IDP   FRZ#SAV  =      APL          FWA OF SAVE AREA FOR (RA.SSW), (FL-2),  001860   DBG=IDP     3947     A
DBG=IDP  *                             AND (FL-1)                       001870   DBG=IDP     3948     A
DBG=IDP   IIF      SPACE  4,8                                           001880   DBG=IDP     3949     A
DBG=IDP  **        IIF - INITIALIZE INTERACTIVE FILES.                  001890   DBG=IDP     3950     A
DBG=IDP  *                                                              001900   DBG=IDP     3951     A
DBG=IDP  *                                                              001910   DBG=IDP     3952     A
DBG=IDP  *         THIS ROUTINE OPENS AND CONNECTS THE INTERACTIVE DEBUG INPUT  001920   DBG=IDP     3953     A
DBG=IDP  *         AND OUTPUT FILES, *F.IDI* AND *F.IDO*.               001930   DBG=IDP     3954     A
DBG=IDP  *                                                              001940   DBG=IDP     3955     A
DBG=IDP  *         ENTRY  NONE                                          001950   DBG=IDP     3956     A
DBG=IDP  *                                                              001960   DBG=IDP     3957     A
DBG=IDP  *         EXIT   NONE                                          001970   DBG=IDP     3958     A
DBG=IDP  *                                                              001980   DBG=IDP     3959     A
DBG=IDP  *         USES   ALL BUT A5,X5,A0,X0                           001990   DBG=IDP     3960     A
DBG=IDP  *                                                              002000   DBG=IDP     3961     A
DBG=IDP  *         CALLS  CON,OPEN                                      002010   DBG=IDP     3962     A
DBG=IDP                                                                 002020   DBG=IDP     3963     A
DBG=IDP                                                                 002030   DBG=IDP     3964     A
DBG=IDP   IIF      SUBR                ** ENTRY/EXIT **                 002040   DBG=IDP     3965     A
DBG=IDP                                                                 002050   DBG=IDP     3966     A
DBG=IDP  *         SET UP *F.IDI - INTERACTIVE DEBUG INPUT FILE*.       002060   DBG=IDP     3967     A
DBG=IDP                                                                 002070   DBG=IDP     3968     A
DBG=IDP            SA1    F.IDI                                         002080   DBG=IDP     3969     A
DBG=IDP            SX2    040021B      (X2) = CIO EOR STATUS            002090   DBG=IDP     3970     A
DBG=IDP            MX0    7*CHAR                                        002100   DBG=IDP     3971     A
DBG=IDP            BX6    X0*X1                                         002110   DBG=IDP     3972     A
DBG=IDP            BX6    X6+X2                                         002120   DBG=IDP     3973     A
DBG=IDP            SA6    A1                                            002130   DBG=IDP     3974     A
DBG=IDP            OPEN   F.IDI,,RCL                                    002140   DBG=IDP     3975     A
DBG=IDP            SA1    F.IDI                                         002150   DBG=IDP     3976     A
DBG=IDP            MX2    0            SET TO *CONNECT*                 002160   DBG=IDP     3977     A
DBG=IDP            RJ     CON          CONNECT F.IDI                    002170   DBG=IDP     3978     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP                                                                  002180   DBG=IDP     3979     A
DBG=IDP   #OS3     IFEQ    .OS,3       IF SCOPE 3                        002190   DBG=IDP     3980     A
DBG=IDP            MI      X1,*+4S15   IF *CON* DETECTED ERROR...        002200   DBG=IDP     3981     A
DBG=IDP   #OS3     ENDIF                                                 002210   DBG=IDP     3982     A
DBG=IDP                                                                  002220   DBG=IDP     3983     A
DBG=IDP   *        SET UP *F.IDO - INTERACTIVE DEBUG OUTPUT FILE*.       002230   DBG=IDP     3984     A
DBG=IDP                                                                  002240   DBG=IDP     3985     A
DBG=IDP            OPEN    F.IDO,,RCL                                    002250   DBG=IDP     3986     A
DBG=IDP            SA1     F.IDO                                         002260   DBG=IDP     3987     A
DBG=IDP            MX2     0           SET TO *CONNECT*                  002270   DBG=IDP     3988     A
DBG=IDP            RJ      CON         CONNECT F.IDO                     002280   DBG=IDP     3989     A
DBG=IDP                                                                  002290   DBG=IDP     3990     A
DBG=IDP   #OS3     IFEQ    .OS,3       IF SCOPE 3                        002300   DBG=IDP     3991     A
DBG=IDP            MI      X1,*+4S15   IF *CON* DETECTED ERROR...        002310   DBG=IDP     3992     A
DBG=IDP   #OS3     ENDIF                                                 002320   DBG=IDP     3993     A
DBG=IDP                                                                  002330   DBG=IDP     3994     A
DBG=IDP            EQ      EXIT.                                         002340   DBG=IDP     3995     A
DBG=IDP    IST     SPACE   4,8                                                    DBG=IDP     3996     A
DBG=IDP   **       IST - INITIALIZE *SET* TABLE.                                  DBG=IDP     3997     A
DBG=IDP   *                                                                       DBG=IDP     3998     A
DBG=IDP   *                                                                       DBG=IDP     3999     A
DBG=IDP   *                THIS ROUTINE SETS UP THE DEFAULT *SET* TABLE VALUES.   DBG=IDP     4000     A
DBG=IDP   *                                                                       DBG=IDP     4001     A
DBG=IDP   *        ENTRY   NONE                                                   DBG=IDP     4002     A
DBG=IDP   *                                                                       DBG=IDP     4003     A
DBG=IDP   *        EXIT    NONE                                                   DBG=IDP     4004     A
DBG=IDP   *                                                                       DBG=IDP     4005     A
DBG=IDP   *        USES    ALL                                                    DBG=IDP     4006     A
DBG=IDP   *                                                                       DBG=IDP     4007     A
DBG=IDP   *        CALLS   ADZ                                                    DBG=IDP     4008     A
DBG=IDP                                                                           DBG=IDP     4009     A
DBG=IDP                                                                           DBG=IDP     4010     A
DBG=IDP    IST     SUBR                ** ENTRY/EXIT **                           DBG=IDP     4011     A
DBG=IDP            SB6     IDPSET      (B6) = FWA OF IDP *SET* TABLE              DBG=IDP     4012     A
DBG=IDP            SA1     =4LTEMP                                                DBG=IDP     4013     A
DBG=IDP            SX6     IDPTMP                                                 DBG=IDP     4014     A
DBG=IDP            BX6     X1+X6       (X6) = 42/0LTEMP,18/FWA OF TEMP AREA       DBG=IDP     4015     A
DBG=IDP            RJ      ADZ         ADD WORD TO IDP TABLE                      DBG=IDP     4016     A
DBG=IDP            SX7     1RP                                                    DBG=IDP     4017     A
DBG=IDP            SX6     IDPPREG                                                DBG=IDP     4018     A
DBG=IDP            LX7     10*CHAR-1*CHAR                                         DBG=IDP     4019     A
DBG=IDP            BX6     X7+X6       (X6) = 42/0LP, 18/ADDR OF PSEUDO P REGISTER DBG=IDP    4020     A
DBG=IDP            RJ      ADZ         ADD WORD TO IDP TABLE                      DBG=IDP     4021     A
DBG=IDP            EQ      EXIT.                                                  DBG=IDP     4022     A
DBG=IDP    LBT     SPACE   4,8                                                    DBG=IDP     4023     A
DBG=IDP   **       LBT - LIST BREAK TABLE.                                        DBG=IDP     4024     A
DBG=IDP   *                                                                       DBG=IDP     4025     A
DBG=IDP   *                                                                       DBG=IDP     4026     A
DBG=IDP   *                LISTS THE CONTENTS OF THE PARALLEL BREAK TABLES        DBG=IDP     4027     A
DBG=IDP   *        *IDPBA/IDPBC* IN THREE FORMS, DEPENDING ON THE TYPE OF BREAK-- DBG=IDP     4028     A
DBG=IDP   *                                                                       DBG=IDP     4029     A
DBG=IDP   *     COL 1         1         1         1        (.=BLANK(55B))       DBG=IDP     4030     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *         .BREAK.AT.NNNNNN.IN.XXXXXXX                                  DBG=IDP    4031    A
DBG=IDP  *         ...........LL.NNNNN..UL.NNNNN..INC.NNNN.CNT.NNNNNN           DBG=IDP    4032    A
DBG=IDP  *                                                                      DBG=IDP    4033    A
DBG=IDP  *         .BRPL..AT.NNNNNN.IN.XXXXXXX....A1.NNNNNN                     DBG=IDP    4034    A
DBG=IDP  *         ...........LL.NNNNN..UL.NNNNN..INC.NNNN.CNT.NNNNNN           DBG=IDP    4035    A
DBG=IDP  *                                                                      DBG=IDP    4036    A
DBG=IDP  *         .BREQ..AT.NNNNNN.IN.XXXXXXX....A1.NNNNNN.A2.NNNNNN           DBG=IDP    4037    A
DBG=IDP  *         ...........LL.NNNNN..UL.NNNNN..INC.NNNN.CNT.NNNNNN           DBG=IDP    4038    A
DBG=IDP  *                                                                      DBG=IDP    4039    A
DBG=IDP  *         ENTRY  NONE                                                  DBG=IDP    4040    A
DBG=IDP  *                                                                      DBG=IDP    4041    A
DBG=IDP  *         EXIT   NONE                                                  DBG=IDP    4042    A
DBG=IDP  *                                                                      DBG=IDP    4043    A
DBG=IDP  *         USES   ALL BUT A0                                            DBG=IDP    4044    A
DBG=IDP  *                                                                      DBG=IDP    4045    A
DBG=IDP  *         CALLS  COD,FRA=,PRIDP,SFN                                    DBG=IDP    4046    A
DBG=IDP                                                                         DBG=IDP    4047    A
DBG=IDP                                                                         DBG=IDP    4048    A
DBG=IDP   LBT      SUBR               ** ENTRY/EXIT **                          DBG=IDP    4049    A
DBG=IDP                                                                         DBG=IDP    4050    A
DBG=IDP  *         SET UP FOR 1ST ENTRY.                                        DBG=IDP    4051    A
DBG=IDP                                                                         DBG=IDP    4052    A
DBG=IDP            SA5    IDPBA                                                 DBG=IDP    4053    A
DBG=IDP            EQ     LBT2                                                  DBG=IDP    4054    A
DBG=IDP                                                                         DBG=IDP    4055    A
DBG=IDP  *         MAIN LOOP.                                                   DBG=IDP    4056    A
DBG=IDP                                                                         DBG=IDP    4057    A
DBG=IDP   LBT1     SA5    A5+B1                                                 DBG=IDP    4058    A
DBG=IDP                                                                         DBG=IDP    4059    A
DBG=IDP   LBT2     ZR     X5,EXIT.    IF FINISHED LISTING ALL BREAKS            DBG=IDP    4060    A
DBG=IDP            SX6    X5+B1                                                 DBG=IDP    4061    A
DBG=IDP            ZR     X6,LBT1     IF THIS ENTRY AVAILABLE (NOT A BREAK)     DBG=IDP    4062    A
DBG=IDP                                                                         DBG=IDP    4063    A
DBG=IDP  *         ASSEMBLE * BREAK AT *.                                       DBG=IDP    4064    A
DBG=IDP                                                                         DBG=IDP    4065    A
DBG=IDP            SX4    X5          (X4) = BREAK ADDRESS                      DBG=IDP    4066    A
DBG=IDP            LX5    6                                                     DBG=IDP    4067    A
DBG=IDP            MX0    -6                                                    DBG=IDP    4068    A
DBG=IDP            BX6    -X0*X5      (X6) = BREAK TYPE CODE                    DBG=IDP    4069    A
DBG=IDP            MX0    7*CHAR                                                DBG=IDP    4070    A
DBG=IDP            SA1    IDPKBR+X6                                             DBG=IDP    4071    A
DBG=IDP            SA6    IDPBTC                                                DBG=IDP    4072    A
DBG=IDP            BX1    X0*X1                                                 DBG=IDP    4073    A
DBG=IDP            RJ     =XSFN       SPACE FILL NAME                           DBG=IDP    4074    A
DBG=IDP            LX6    -CHAR                                                 DBG=IDP    4075    A
DBG=IDP            BX6    X0*X6                                                 DBG=IDP    4076    A
DBG=IDP            SX7    3RAT                                                  DBG=IDP    4077    A
DBG=IDP            BX6    X6+X7       (X6) = .BREAK.AT.   (.=BLANK(55B))        DBG=IDP    4078    A
DBG=IDP            SA6    SNAPLNE                                               DBG=IDP    4079    A
DBG=IDP                                                                         DBG=IDP    4080    A
DBG=IDP  *         ASSEMBLE *NNNNNN IN XXXXXXX   *.                             DBG=IDP    4081    A
DBG=IDP                                                                         DBG=IDP    4082    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            SX1    X4                                                    DBG=IDP    4083    A
DBG=IDP                                                                         DBG=IDP    4084    A
DBG=IDP   #FRA     IF     DEF,FRA=                                              DBG=IDP    4085    A
DBG=IDP            SB7    LBT3        (B7) = RETURN ADDR FOR *FRA=*             DBG=IDP    4086    A
DBG=IDP            EQ     =XFRA=      FIND RELATIVE ADDRESS                     DBG=IDP    4087    A
DBG=IDP                                                                         DBG=IDP    4088    A
DBG=IDP   LBT3     BSS    0                                                     DBG=IDP    4089    A
DBG=IDP                                                                         DBG=IDP    4090    A
DBG=IDP   #FRA     ELSE                                                         DBG=IDP    4091    A
DBG=IDP            RJ     =XCOD       CONVERT BINARY TO OCTAL DPC               DBG=IDP    4092    A
DBG=IDP            BX6    X4                                                    DBG=IDP    4093    A
DBG=IDP            MX7    0                                                     DBG=IDP    4094    A
DBG=IDP   #FRA     ENDIF                                                        DBG=IDP    4095    A
DBG=IDP                                                                         DBG=IDP    4096    A
DBG=IDP            SA6    A6+B1                                                 DBG=IDP    4097    A
DBG=IDP            SA7    A6+B1                                                 DBG=IDP    4098    A
DBG=IDP            SA2    IDPBTC                                                DBG=IDP    4099    A
DBG=IDP            ZR     X2,LBT5     IF *BREAK ADDR,LL,UL,INC* FORM            DBG=IDP    4100    A
DBG=IDP            BX1    X7                                                    DBG=IDP    4101    A
DBG=IDP            RJ     SFN         SPACE FILL NAME                           DBG=IDP    4102    A
DBG=IDP            SA6    A6+B1       RE-STORE BLANK FILLED *XXXXXXX   *        DBG=IDP    4103    A
DBG=IDP                                                                         DBG=IDP    4104    A
DBG=IDP   *        ASSEMBLE * A1 NNNNNN*.                                       DBG=IDP    4105    A
DBG=IDP                                                                         DBG=IDP    4106    A
DBG=IDP            LX5    18                                                    DBG=IDP    4107    A
DBG=IDP            SX1    X5                                                    DBG=IDP    4108    A
DBG=IDP            RJ     =XCOD       CONVERT BINARY TO OCTAL DPC               DBG=IDP    4109    A
DBG=IDP            SA2    IDPBTC                                                DBG=IDP    4110    A
DBG=IDP            SX1    3R A1                                                 DBG=IDP    4111    A
DBG=IDP            MX0    -7*CHAR                                               DBG=IDP    4112    A
DBG=IDP            BX6    -X0*X6                                                DBG=IDP    4113    A
DBG=IDP            LX1    10*CHAR-3*CHAR                                        DBG=IDP    4114    A
DBG=IDP            BX1    X0*X1                                                 DBG=IDP    4115    A
DBG=IDP            BX6    X6+X1       (X6) = .A1.NNNNNN   (.=BLANK(55B))        DBG=IDP    4116    A
DBG=IDP            SA6    A6+B1                                                 DBG=IDP    4117    A
DBG=IDP            SX2    X2-BTC=EQ                                             DBG=IDP    4118    A
DBG=IDP            MI     X2,LBT4     IF *BRPL ADDR,A1,LL,UL,INC* FORM          DBG=IDP    4119    A
DBG=IDP                                                                         DBG=IDP    4120    A
DBG=IDP   *        ASSEMBLE * A2 NNNNNN*.                                       DBG=IDP    4121    A
DBG=IDP                                                                         DBG=IDP    4122    A
DBG=IDP            LX5    18                                                    DBG=IDP    4123    A
DBG=IDP            SX1    X5                                                    DBG=IDP    4124    A
DBG=IDP            RJ     =XCOD       CONVERT BINARY TO OCTAL DPC               DBG=IDP    4125    A
DBG=IDP            SX1    3R A2                                                 DBG=IDP    4126    A
DBG=IDP            MX0    -7*CHAR                                               DBG=IDP    4127    A
DBG=IDP            BX6    -X0*X6                                                DBG=IDP    4128    A
DBG=IDP            LX1    10*CHAR-3*CHAR                                        DBG=IDP    4129    A
DBG=IDP            BX1    X0*X1                                                 DBG=IDP    4130    A
DBG=IDP            BX6    X6+X1       (X6) = .A2.NNNNNN   (.=BLANK(55B))        DBG=IDP    4131    A
DBG=IDP            SA6    A6+B1                                                 DBG=IDP    4132    A
DBG=IDP                                                                         DBG=IDP    4133    A
DBG=IDP   LBT4     MX6    0                                                     DBG=IDP    4134    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            SA6    A6+B1       MARK EOL                                    DBG=IDP    4135    A
DBG=IDP                                                                            DBG=IDP    4136    A
DBG=IDP   LBT5     PRIDP  SNAPLNE                                                  DBG=IDP    4137    A
DBG=IDP                                                                            DBG=IDP    4138    A
DBG=IDP   *        ASSEMBLE *          LL NNNNN *.                                DBG=IDP    4139    A
DBG=IDP                                                                            DBG=IDP    4140    A
DBG=IDP            SA1    =10H                                                    DBG=IDP    4141    A
DBG=IDP            SB7    A5-IDPBA                                                DBG=IDP    4142    A
DBG=IDP            SB6    B7+B7                                                   DBG=IDP    4143    A
DBG=IDP            SA2    IDPBC+B6+1  (X2) = 2ND WORD OF *IDPBC* ENTRY            DBG=IDP    4144    A
DBG=IDP            BX6    X1                                                      DBG=IDP    4145    A
DBG=IDP            BX5    X2                                                      DBG=IDP    4146    A
DBG=IDP            LX5    15                                                      DBG=IDP    4147    A
DBG=IDP            MX0    -15                                                     DBG=IDP    4148    A
DBG=IDP            BX1    -X0*X5                                                  DBG=IDP    4149    A
DBG=IDP            SA6    SNAPLNE                                                 DBG=IDP    4150    A
DBG=IDP            RJ     =XCOD       CONVERT BINARY TO OCTAL DPC                 DBG=IDP    4151    A
DBG=IDP            SX1    3R LL                                                   DBG=IDP    4152    A
DBG=IDP            LX4    -4*CHAR     LEFT JUSTIFY NR TO BIT 35                   DBG=IDP    4153    A
DBG=IDP            BX6    X4          (X6) = ....NNNNN.   (.=BLANK(55B))          DBG=IDP    4154    A
DBG=IDP            MX0    -7*CHAR                                                 DBG=IDP    4155    A
DBG=IDP            LX1    10*CHAR-3*CHAR                                          DBG=IDP    4156    A
DBG=IDP            BX1    X0*X1                                                   DBG=IDP    4157    A
DBG=IDP            BX6    -X0*X6                                                  DBG=IDP    4158    A
DBG=IDP            BX6    X6+X1       (X6) = .LL.NNNNN.   (.=BLANK(55B))          DBG=IDP    4159    A
DBG=IDP            SA6    A6+B1                                                   DBG=IDP    4160    A
DBG=IDP                                                                            DBG=IDP    4161    A
DBG=IDP   *        ASSEMBLE * UL NNNNN *.                                         DBG=IDP    4162    A
DBG=IDP                                                                            DBG=IDP    4163    A
DBG=IDP            LX5    15                                                      DBG=IDP    4164    A
DBG=IDP            MX0    -15                                                     DBG=IDP    4165    A
DBG=IDP            BX1    -X0*X5      (X1) = UL                                   DBG=IDP    4166    A
DBG=IDP            RJ     =XCOD       CONVERT BINARY TO OCTAL DPC                 DBG=IDP    4167    A
DBG=IDP            SX1    3R UL                                                   DBG=IDP    4168    A
DBG=IDP            LX4    -4*CHAR                                                 DBG=IDP    4169    A
DBG=IDP            BX6    X4          (X6) = ....NNNNN.   (.=BLANK(55B))          DBG=IDP    4170    A
DBG=IDP            MX0    -7*CHAR                                                 DBG=IDP    4171    A
DBG=IDP            BX6    -X0*X6                                                  DBG=IDP    4172    A
DBG=IDP            LX1    10*CHAR-3*CHAR                                          DBG=IDP    4173    A
DBG=IDP            BX1    X0*X1                                                   DBG=IDP    4174    A
DBG=IDP            BX6    X6+X1       (X6) = .UL.NNNNN.   (.=BLANK(55B))          DBG=IDP    4175    A
DBG=IDP            SA6    A6+B1                                                   DBG=IDP    4176    A
DBG=IDP                                                                            DBG=IDP    4177    A
DBG=IDP   *        ASSEMBLE * INC NNNN *.                                         DBG=IDP    4178    A
DBG=IDP                                                                            DBG=IDP    4179    A
DBG=IDP            LX5    12                                                      DBG=IDP    4180    A
DBG=IDP            MX0    -12                                                     DBG=IDP    4181    A
DBG=IDP            BX1    -X0*X5      (X1) = INC                                  DBG=IDP    4182    A
DBG=IDP            RJ     =XCOD       CONVERT BINARY TO OCTAL DPC                 DBG=IDP    4183    A
DBG=IDP            SA1    =4L INC                                                 DBG=IDP    4184    A
DBG=IDP            LX4    -5*CHAR                                                 DBG=IDP    4185    A
DBG=IDP            BX6    X4          (X6) = .....NNNN.   (.=BLANK(55B))          DBG=IDP    4186    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            MX0    -6*CHAR                                                DBG=IDP     4187     A
DBG=IDP            BX6    -X0*X6
DBG=IDP            BX6    X6+X1       (X6) = .INC.NNNN.   (.=BLANK(55B))         DBG=IDP     4188     A
DBG=IDP            SA6    A6+B1                                                  DBG=IDP     4189     A
DBG=IDP                                                                           DBG=IDP     4190     A
DBG=IDP  *         ASSEMBLE *CNT NNNNNN*.                                         DBG=IDP     4191     A
DBG=IDP                                                                           DBG=IDP     4192     A
DBG=IDP            LX5    18                                                     DBG=IDP     4193     A
DBG=IDP            SX1    X5                                                     DBG=IDP     4194     A
DBG=IDP            RJ     =XCOD       CONVERT BINARY TO OCTAL DPC                DBG=IDP     4195     A
DBG=IDP            SX1    3RCNT                                                  DBG=IDP     4196     A
DBG=IDP            MX0    -7*CHAR                                                DBG=IDP     4197     A
DBG=IDP            BX6    -X0*X6                                                 DBG=IDP     4198     A
DBG=IDP            LX1    10*CHAR-3*CHAR                                         DBG=IDP     4199     A
DBG=IDP            BX1    X0*X1                                                  DBG=IDP     4200     A
DBG=IDP            BX6    X6+X1       (X6) = CNT.NNNNNN   (.=BLANK(55B))         DBG=IDP     4201     A
DBG=IDP            MX7    0                                                      DBG=IDP     4202     A
DBG=IDP            SA6    A6+B1                                                  DBG=IDP     4203     A
DBG=IDP            SA7    A6+B1       MARK EOL                                   DBG=IDP     4204     A
DBG=IDP            PRIDP  SNAPLNE                                                DBG=IDP     4205     A
DBG=IDP            EQ     LBT1                                                   DBG=IDP     4206     A
DBG=IDP   LST      SPACE  4,8                                                    DBG=IDP     4207     A
DBG=IDP  **        LST - LIST *SET* NAME TABLE.                                  DBG=IDP     4208     A
DBG=IDP  *                                                                       DBG=IDP     4209     A
DBG=IDP  *                                                                       DBG=IDP     4210     A
DBG=IDP  *         ENTRY  (A5,X5) =   ADDR + CNTS OF 1ST *SET* TABLE ENTRY       DBG=IDP     4211     A
DBG=IDP  *                                                                       DBG=IDP     4212     A
DBG=IDP  *         EXIT   NONE                                                   DBG=IDP     4213     A
DBG=IDP  *                                                                       DBG=IDP     4214     A
DBG=IDP  *         USES   ALL                                                    DBG=IDP     4215     A
DBG=IDP  *                                                                       DBG=IDP     4216     A
DBG=IDP  *         CALLS  COD,FRA=,PRIDP,SFN                                     DBG=IDP     4217     A
DBG=IDP                                                                          DBG=IDP     4218     A
DBG=IDP                                                                          DBG=IDP     4219     A
DBG=IDP   LST      SUBR               ** ENTRY/EXIT **                           DBG=IDP     4220     A
DBG=IDP                                                                          DBG=IDP     4221     A
DBG=IDP   LST2     ZR     X5,EXIT.    IF FINISHED ALL SET NAMES                  DBG=IDP     4222     A
DBG=IDP            SX1    B1                                                     DBG=IDP     4223     A
DBG=IDP            IX6    X5+X1                                                  DBG=IDP     4224     A
DBG=IDP            ZR     X6,LST4     IF THIS ENTRY IS AVAILABLE (NOT A SET)     DBG=IDP     4225     A
DBG=IDP            MX0    7*CHAR                                                 DBG=IDP     4226     A
DBG=IDP            BX1    X0*X5       (X1) = 0LNAME                              DBG=IDP     4227     A
DBG=IDP            RJ     =XSFN       SPACE FILL NAME                            DBG=IDP     4228     A
DBG=IDP            LX6    -CHAR                                                  DBG=IDP     4229     A
DBG=IDP            SA6    SNAPLNE                                                DBG=IDP     4230     A
DBG=IDP            BX1    -X0*X5                                                 DBG=IDP     4231     A
DBG=IDP            RJ     =XCOD       CONVERT BINARY TO OCTAL DPC-- SET NAME VALU DBG=IDP     4232     A
DBG=IDP            LX6    4*CHAR                                                 DBG=IDP     4233     A
DBG=IDP            SA6    A6+B1                                                  DBG=IDP     4234     A
DBG=IDP                                                                          DBG=IDP     4235     A
DBG=IDP   #FRA     IF     DEF,FRA=                                               DBG=IDP     4236     A
DBG=IDP            BX1    -X0*X5                                                 DBG=IDP     4237     A
DBG=IDP                                                                          DBG=IDP     4238     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            SB7    LST3        (B7) = RETURN ADDR FOR *FRA=*                    DBG=IDP    4239    A
DBG=IDP            EQ     =XFRA=      FIND RELATIVE ADDRESS                            DBG=IDP    4240    A
DBG=IDP                                                                                DBG=IDP    4241    A
DBG=IDP   LST3     SA6    A6+B1                                                        DBG=IDP    4242    A
DBG=IDP            SA7    A6+B1                                                        DBG=IDP    4243    A
DBG=IDP                                                                                DBG=IDP    4244    A
DBG=IDP   #FRA     ELSE                                                                DBG=IDP    4245    A
DBG=IDP            MX6    0                                                            DBG=IDP    4246    A
DBG=IDP            SA6    A6+B1                                                        DBG=IDP    4247    A
DBG=IDP   #FRA     ENDIF                                                               DBG=IDP    4248    A
DBG=IDP                                                                                DBG=IDP    4249    A
DBG=IDP            PRIDP  SNAPLNE                                                      DBG=IDP    4250    A
DBG=IDP                                                                                DBG=IDP    4251    A
DBG=IDP   LST4     SA5    A5+B1                                                        DBG=IDP    4252    A
DBG=IDP            EQ     LST2                                                         DBG=IDP    4253    A
DBG=IDP   LXT      SPACE  4,8                                                          DBG=IDP    4254    A
DBG=IDP  **        LXT - LIST XFER TABLE.                                              DBG=IDP    4255    A
DBG=IDP  *                                                                             DBG=IDP    4256    A
DBG=IDP  *                                                                             DBG=IDP    4257    A
DBG=IDP  *                LISTS THE MOST RECENT TRANSFERS OF CONTROL TO OCCUR          DBG=IDP    4258    A
DBG=IDP  *         DURING STEP MODE (CONTAINED IN *IDPXFT*).                           DBG=IDP    4259    A
DBG=IDP  *                                                                             DBG=IDP    4260    A
DBG=IDP  *         *IDPXFT* FORMAT --                                                  DBG=IDP    4261    A
DBG=IDP  *                                                                             DBG=IDP    4262    A
DBG=IDP  *         VFD    60/-1       UNUSED ENTRY (I.E. XFT HAS NOT FILLED YET)      DBG=IDP    4263    A
DBG=IDP  *          .                                                                  DBG=IDP    4264    A
DBG=IDP  *          .                                                                  DBG=IDP    4265    A
DBG=IDP  *         VFD    30/POS COUNTER,30/P REGISTER OF XFER INSTRUCTION            DBG=IDP    4266    A
DBG=IDP  *          .                                                                  DBG=IDP    4267    A
DBG=IDP  *         VFD    60/0        END OF TABLE MARK                               DBG=IDP    4268    A
DBG=IDP  *                                                                             DBG=IDP    4269    A
DBG=IDP  *         ENTRY  (X6) = NR OF ENTRIES TO LIST (BEGINNING W/ MOST RECENT)     DBG=IDP    4270    A
DBG=IDP  *                                                                             DBG=IDP    4271    A
DBG=IDP  *         EXIT   NONE                                                         DBG=IDP    4272    A
DBG=IDP  *                                                                             DBG=IDP    4273    A
DBG=IDP  *         USES   ALL                                                          DBG=IDP    4274    A
DBG=IDP  *                                                                             DBG=IDP    4275    A
DBG=IDP  *         CALLS  DAZ                                                          DBG=IDP    4276    A
DBG=IDP                                                                                DBG=IDP    4277    A
DBG=IDP                                                                                DBG=IDP    4278    A
DBG=IDP   LXT      SUBR               ** ENTRY/EXIT **                                 DBG=IDP    4279    A
DBG=IDP                                                                                DBG=IDP    4280    A
DBG=IDP  *         INITIALIZATION.                                                     DBG=IDP    4281    A
DBG=IDP                                                                                DBG=IDP    4282    A
DBG=IDP            SX1    IDPXFT+L.XFT                                                 DBG=IDP    4283    A
DBG=IDP            SX7    X6-L.XFT-1                                                   DBG=IDP    4284    A
DBG=IDP            MI     X7,LXT2     IF NOT ASKING FOR TOO MANY ENTRIES               DBG=IDP    4285    A
DBG=IDP            SX6    L.XFT                                                        DBG=IDP    4286    A
DBG=IDP                                                                                DBG=IDP    4287    A
DBG=IDP   LXT2     IX7    X1-X6                                                        DBG=IDP    4288    A
DBG=IDP            NO                                                                  DBG=IDP    4289    A
DBG=IDP            SA7    LXT=NXT                                                      DBG=IDP    4290    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP                                                                    DBG=IDP     4291     A
DBG=IDP  *         ** MAIN LOOP NODE **                                    DBG=IDP     4292     A
DBG=IDP                                                                    DBG=IDP     4293     A
DBG=IDP   LXT3     SA2    LXT=NXT                                          DBG=IDP     4294     A
DBG=IDP            SA3    X2          (X3) = NEXT *XFT* ENTRY              DBG=IDP     4295     A
DBG=IDP            SX7    X2+B1                                            DBG=IDP     4296     A
DBG=IDP            SX4    X3+B1                                            DBG=IDP     4297     A
DBG=IDP            SA7    A2                                               DBG=IDP     4298     A
DBG=IDP            ZR     X3,EXIT.    IF HIT END OF TABLE MARK, DONE...    DBG=IDP     4299     A
DBG=IDP            ZR     X4,LXT3     IF HIT AVAILABLE ENTRY (I.E. *XFT* WAS NOT  DBG=IDP     4300     A
DBG=IDP  *                               FULL TO BEGIN WITH)               DBG=IDP     4301     A
DBG=IDP            SA5    X3          (A5,X5) = A+C OF XFER INSTRUCTION    DBG=IDP     4302     A
DBG=IDP            AX3    30                                               DBG=IDP     4303     A
DBG=IDP            SB4    X3          (B4) = POS COUNTER                   DBG=IDP     4304     A
DBG=IDP            SA7    IDPFAD      SET TO *FORCE OUT ADDR UNCONDITIONALLY*  DBG=IDP     4305     A
DBG=IDP            RJ     DAZ         DEASSEMBLE XFER INSTRUCTION          DBG=IDP     4306     A
DBG=IDP            EQ     LXT3                                             DBG=IDP     4307     A
DBG=IDP                                                                    DBG=IDP     4308     A
DBG=IDP                                                                    DBG=IDP     4309     A
DBG=IDP   LXT=NXT  =      APL         SAVED ADDR OF NEXT *XFT* ENTRY       DBG=IDP     4310     A
DBG=IDP   PAS      SPACE  4,8                                              DBG=IDP     4311     A
DBG=IDP  **        PAS - PARSE SUBEXPRESSION.                              DBG=IDP     4312     A
DBG=IDP  *                                                                 DBG=IDP     4313     A
DBG=IDP  *                THIS ROUTINE PARSES AN EXPRESSION THAT IS TERMINATED  DBG=IDP     4314     A
DBG=IDP  *         BY A -,- OR -EOS-, AND RETURNS THE VALUE OF THE EXPRESSION  DBG=IDP     4315     A
DBG=IDP  *         IN THE ACCUMULATOR *IDPACC*.                            DBG=IDP     4316     A
DBG=IDP  *                                                                 DBG=IDP     4317     A
DBG=IDP  *         ENTRY  (A5+1,X5)=  ADDR + CONTENTS OF 1ST TOKEN IN EXPRESSION  DBG=IDP     4318     A
DBG=IDP  *                                                                 DBG=IDP     4319     A
DBG=IDP  *         EXIT   (IDPACC) =  BINARY VALUE OF EXPRESSION           DBG=IDP     4320     A
DBG=IDP  *                (X1)     =  (IDPACC)                             DBG=IDP     4321     A
DBG=IDP  *                (X3)     =  .ZR. IF EXPRESSION WAS NULL, ELSE .NZ.  DBG=IDP     4322     A
DBG=IDP  *                (A5,X5)  =  ADDR + CNTS OF TOKEN THAT TERMINATED PARSE  DBG=IDP     4323     A
DBG=IDP  *                (X6)     =  (X1) = (IDPACC)                      DBG=IDP     4324     A
DBG=IDP  *                                                                 DBG=IDP     4325     A
DBG=IDP  *         USES   ALL                                              DBG=IDP     4326     A
DBG=IDP  *                                                                 DBG=IDP     4327     A
DBG=IDP  *         CALLS  DXB,FAA(IF DEF),IDP=ER,SKT                       DBG=IDP     4328     A
DBG=IDP                                                                    DBG=IDP     4329     A
DBG=IDP                                                                    DBG=IDP     4330     A
DBG=IDP   PAS      SUBR               ** ENTRY/EXIT **                     DBG=IDP     4331     A
DBG=IDP                                                                    DBG=IDP     4332     A
DBG=IDP  *         INITIALIZATION.                                         DBG=IDP     4333     A
DBG=IDP                                                                    DBG=IDP     4334     A
DBG=IDP            MX6    0                                                DBG=IDP     4335     A
DBG=IDP            MX7    -1                                               DBG=IDP     4336     A
DBG=IDP            SA6    IDPACC      CLEAR ACCUMULATOR                    DBG=IDP     4337     A
DBG=IDP            SA6    IDPSGN      SET TO *SIGN IS PLUS*                DBG=IDP     4338     A
DBG=IDP            SA6    IDPIND      SET TO *DIRECT ADDRESSING*           DBG=IDP     4339     A
DBG=IDP            SA7    IDPNUL      SET TO *NULL EXPRESSION SO FAR*      DBG=IDP     4340     A
DBG=IDP                                                                    DBG=IDP     4341     A
DBG=IDP  *         MAIN LOOP NODE.                                         DBG=IDP     4342     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP                                                                    DBG=IDP     4343     A
DBG=IDP   PS=ALL   SA5    A5+B1       (X5) = NEXT TOKEN TO PARSE           DBG=IDP     4344     A
DBG=IDP            SA4    IDPNUL                                           DBG=IDP     4345     A
DBG=IDP            SX7    X4+B1                                            DBG=IDP     4346     A
DBG=IDP            SA7    A4                                               DBG=IDP     4347     A
DBG=IDP            ZR     X5,PS=END   IF EOS TOKEN                         DBG=IDP     4348     A
DBG=IDP            MX0    -CHAR                                            DBG=IDP     4349     A
DBG=IDP            =X6    X5-O.VAR                                         DBG=IDP     4350     A
DBG=IDP            =X7    X5-O.CONS                                        DBG=IDP     4351     A
DBG=IDP            ZR     X6,PS=NAM   IF NAME TOKEN                        DBG=IDP     4352     A
DBG=IDP            ZR     X7,PS=CON   IF CONSTANT TOKEN                    DBG=IDP     4353     A
DBG=IDP            SX6    X5-O.SEP                                         DBG=IDP     4354     A
DBG=IDP            NZ     X6,IDP.ER   IF NOT SEPARATOR TOKEN, ERR...       DBG=IDP     4355     A
DBG=IDP                                                                    DBG=IDP     4356     A
DBG=IDP   *        HERE TO PROCESS A SEPARATOR TOKEN -- *,+-               DBG=IDP     4357     A
DBG=IDP                                                                    DBG=IDP     4358     A
DBG=IDP            LX5    1*CHAR-10*CHAR                                   DBG=IDP     4359     A
DBG=IDP            BX5    -X0*X5                                           DBG=IDP     4360     A
DBG=IDP            SX6    X5-1R,                                           DBG=IDP     4361     A
DBG=IDP            SX7    X5-1R*                                           DBG=IDP     4362     A
DBG=IDP            ZR     X6,PS=END   IF , TOKEN                           DBG=IDP     4363     A
DBG=IDP            ZR     X7,PS=STR   IF * TOKEN                           DBG=IDP     4364     A
DBG=IDP            SX6    X5-1R+                                           DBG=IDP     4365     A
DBG=IDP            SX7    X5-1R-                                           DBG=IDP     4366     A
DBG=IDP            ZR     X6,PS=PL    IF + TOKEN                           DBG=IDP     4367     A
DBG=IDP            ZR     X7,PS=MI    IF - TOKEN                           DBG=IDP     4368     A
DBG=IDP            EQ     IDP.ER      ERR - *UNKNOWN TOKEN*                DBG=IDP     4369     A
DBG=IDP                                                                    DBG=IDP     4370     A
DBG=IDP   *        HERE IF NAME TOKEN.                                     DBG=IDP     4371     A
DBG=IDP                                                                    DBG=IDP     4372     A
DBG=IDP   PS=NAM   SA1    IDPBAX                                           DBG=IDP     4373     A
DBG=IDP            BX2    X5                                               DBG=IDP     4374     A
DBG=IDP            MX0    -CHAR                                            DBG=IDP     4375     A
DBG=IDP            LX2    CHAR                                             DBG=IDP     4376     A
DBG=IDP            BX6    -X0*X2                                           DBG=IDP     4377     A
DBG=IDP            SB2    X6          (B2) = REGISTER TYPE (BAX)           DBG=IDP     4378     A
DBG=IDP            LX7    X1,B2                                            DBG=IDP     4379     A
DBG=IDP            PL     X7,PS=NAM2  IF NAME DOES NOT BEGIN WITH B,A,OR X DBG=IDP     4380     A
DBG=IDP                                                                    DBG=IDP     4381     A
DBG=IDP   *        CHECK FOR REGISTER NAME.                                DBG=IDP     4382     A
DBG=IDP                                                                    DBG=IDP     4383     A
DBG=IDP            MX1    1R7-1R0+1                                        DBG=IDP     4384     A
DBG=IDP            LX1    -1R0        (X1) = (0-7) SHIFT MASK              DBG=IDP     4385     A
DBG=IDP            LX2    CHAR                                             DBG=IDP     4386     A
DBG=IDP            BX6    -X0*X2                                           DBG=IDP     4387     A
DBG=IDP            SB3    X6          (B3) = REGISTER NR (0-7) DPC         DBG=IDP     4388     A
DBG=IDP            LX7    X1,B3                                            DBG=IDP     4389     A
DBG=IDP            PL     X7,PS=NAM2  IF 2ND LETTER OF NAME IS NOT (0-7)   DBG=IDP     4390     A
DBG=IDP            LX2    CHAR                                             DBG=IDP     4391     A
DBG=IDP            BX7    -X0*X2                                           DBG=IDP     4392     A
DBG=IDP            NZ     X7,PS=NAM2  IF 3RD CHAR IS NON-ZERO, NOT A REGISTER DBG=IDP  4393     A
DBG=IDP            SB5    1RB                                              DBG=IDP     4394     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            SB6    000B        (B6) = 0TR, WHERE T=0 (B REG)          DBG=IDP    4395    A
DBG=IDP            EQ     B2,B5,PS=NAM1     IF B REGISTER                    DBG=IDP    4396    A
DBG=IDP            SB5    1RA                                                DBG=IDP    4397    A
DBG=IDP            SB6    010B        (B6) = 0TR, WHERE T=1 (A REG)          DBG=IDP    4398    A
DBG=IDP            EQ     B2,B5,PS=NAM1     IF A REGISTER                    DBG=IDP    4399    A
DBG=IDP            SB6    020B        (B6) = 0TR, WHERE T=2 (X REG)          DBG=IDP    4400    A
DBG=IDP                                                                     DBG=IDP    4401    A
DBG=IDP   PS=NAM1  SB7    X6-1R0      (B7) = REGISTER NR IN BINARY           DBG=IDP    4402    A
DBG=IDP            SB6    B6+B7       (B6) = 0TR                             DBG=IDP    4403    A
DBG=IDP            SX6    B6+SV=B     (X6) = ADDR OF SAVED 0TR REGISTER      DBG=IDP    4404    A
DBG=IDP            EQ     PS=ACC                                             DBG=IDP    4405    A
DBG=IDP                                                                     DBG=IDP    4406    A
DBG=IDP  *         CHECK FOR DECK NAME.                                     DBG=IDP    4407    A
DBG=IDP                                                                     DBG=IDP    4408    A
DBG=IDP   PS=NAM2  MX0    7*CHAR                                            DBG=IDP    4409    A
DBG=IDP            BX1    X0*X5       (X1) = 0LNAME                          DBG=IDP    4410    A
DBG=IDP                                                                     DBG=IDP    4411    A
DBG=IDP   #FAA     IF     DEF,FAA=                                           DBG=IDP    4412    A
DBG=IDP            RJ     =XFAA=      FIND ABSOLUTE ADDRESS                  DBG=IDP    4413    A
DBG=IDP            PL     X6,PS=ACC   IF A FIND, ADJUST ACCUMULATOR...       DBG=IDP    4414    A
DBG=IDP   #FAA     ENDIF                                                      DBG=IDP    4415    A
DBG=IDP                                                                     DBG=IDP    4416    A
DBG=IDP  *         HERE TO SEARCH USER *SET* TABLE.                         DBG=IDP    4417    A
DBG=IDP                                                                     DBG=IDP    4418    A
DBG=IDP            SB6    IDPSET      (B6) = FWA OF *SET* TABLE              DBG=IDP    4419    A
DBG=IDP            RJ     SKT         SEARCH SET TABLE FOR NAME              DBG=IDP    4420    A
DBG=IDP            ZR     X2,IDP.ER   IF NO FIND                             DBG=IDP    4421    A
DBG=IDP            SX6    X2+                                                DBG=IDP    4422    A
DBG=IDP            EQ     PS=ACC                                             DBG=IDP    4423    A
DBG=IDP                                                                     DBG=IDP    4424    A
DBG=IDP  *         HERE IF CONSTANT TOKEN.                                  DBG=IDP    4425    A
DBG=IDP                                                                     DBG=IDP    4426    A
DBG=IDP   PS=CON   MX0    7*CHAR                                            DBG=IDP    4427    A
DBG=IDP            BX5    X0*X5       (X5) = 42/0LCONSTANT(DPC), 18/0        DBG=IDP    4428    A
DBG=IDP            SB7    B0          SET TO *MODE IS OCTAL*                 DBG=IDP    4429    A
DBG=IDP            RJ     =XDXB       CONVERT DPC TO BINARY                  DBG=IDP    4430    A
DBG=IDP            NZ     X4,IDP.ER   IF ERROR IN ASSEMBLY                   DBG=IDP    4431    A
DBG=IDP            EQ     PS=ACC      ADJUST ACCUMULATOR...                  DBG=IDP    4432    A
DBG=IDP                                                                     DBG=IDP    4433    A
DBG=IDP  *         HERE IF * TOKEN.                                         DBG=IDP    4434    A
DBG=IDP                                                                     DBG=IDP    4435    A
DBG=IDP   PS=STR   SA1    IDPIND                                             DBG=IDP    4436    A
DBG=IDP            SX6    X1+B1       (X6) = LEVEL OF INDIRECT ADDRESSING    DBG=IDP    4437    A
DBG=IDP            SA6    A1                                                 DBG=IDP    4438    A
DBG=IDP            EQ     PS=ALL      RETURN TO MAIN LOOP.. .                DBG=IDP    4439    A
DBG=IDP                                                                     DBG=IDP    4440    A
DBG=IDP  *         HERE IF + TOKEN .                                        DBG=IDP    4441    A
DBG=IDP                                                                     DBG=IDP    4442    A
DBG=IDP   PS=PL    =      PS=ALL                                             DBG=IDP    4443    A
DBG=IDP                                                                     DBG=IDP    4444    A
DBG=IDP  *         HERE IF - TOKEN.                                         DBG=IDP    4445    A
DBG=IDP                                                                     DBG=IDP    4446    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP   PS=MI    SA1    IDPSGN                                                DBG=IDP    4447    A
DBG=IDP            MX6    60                                                    DBG=IDP    4448    A
DBG=IDP            BX6    X1-X6                                                 DBG=IDP    4449    A
DBG=IDP            SA6    A1                                                    DBG=IDP    4450    A
DBG=IDP            EQ     PS=ALL      RETURN TO MAIN LOOP...                    DBG=IDP    4451    A
DBG=IDP                                                                         DBG=IDP    4452    A
DBG=IDP   *        HERE TO ADJUST ACCUMULATOR.                                  DBG=IDP    4453    A
DBG=IDP                                                                         DBG=IDP    4454    A
DBG=IDP   PS=ACC   SA1    IDPIND                                                DBG=IDP    4455    A
DBG=IDP            SX7    X1-1                                                  DBG=IDP    4456    A
DBG=IDP            MI     X7,PS=ACC2  IF NO MORE LEVELS OF INDIRECT ADDR        DBG=IDP    4457    A
DBG=IDP            SA7    A1                                                    DBG=IDP    4458    A
DBG=IDP            SB2    X6                                                    DBG=IDP    4459    A
DBG=IDP            RJ     CHK         CHECK CM ADDR                             DBG=IDP    4460    A
DBG=IDP            MI     B2,IDP.ER   IF ADDR IS BAD                            DBG=IDP    4461    A
DBG=IDP            SA2    B2          (X2) = NEW ADDRESS                        DBG=IDP    4462    A
DBG=IDP            BX6    X2                                                    DBG=IDP    4463    A
DBG=IDP            EQ     PS=ACC                                                DBG=IDP    4464    A
DBG=IDP                                                                         DBG=IDP    4465    A
DBG=IDP   PS=ACC2  SA1    IDPACC                                                DBG=IDP    4466    A
DBG=IDP            SA2    IDPSGN                                                DBG=IDP    4467    A
DBG=IDP            BX3    X6-X2       (X3) = +X6 IF +, -X6 IF -                 DBG=IDP    4468    A
DBG=IDP            IX7    X1+X3                                                 DBG=IDP    4469    A
DBG=IDP            MX6    0                                                     DBG=IDP    4470    A
DBG=IDP            SA7    A1                                                    DBG=IDP    4471    A
DBG=IDP            SA6    A2          SET TO *DEFAULT SIGN IS PLUS*             DBG=IDP    4472    A
DBG=IDP            EQ     PS=ALL      RETURN TO MAIN LOOP ...                   DBG=IDP    4473    A
DBG=IDP                                                                         DBG=IDP    4474    A
DBG=IDP   *        HERE IF END OF SUBEXPRESSION SENSED.                         DBG=IDP    4475    A
DBG=IDP                                                                         DBG=IDP    4476    A
DBG=IDP   PS=END   SA1    IDPACC                                                DBG=IDP    4477    A
DBG=IDP            SA3    IDPNUL                                                DBG=IDP    4478    A
DBG=IDP            BX6    X1                                                    DBG=IDP    4479    A
DBG=IDP            EQ     EXIT.                                                 DBG=IDP    4480    A
DBG=IDP   PAT      SPACE  4,8                                          000580    DBG=IDP    4481    A
DBG=IDP   **       PAT - PARSE *FWA,LWA,LEN* TRIPLE.                   000590    DBG=IDP    4482    A
DBG=IDP   *                                                            000600    DBG=IDP    4483    A
DBG=IDP   *                                                            000610    DBG=IDP    4484    A
DBG=IDP   *        THIS ROUTINE CONTROLS THE PARSING OF THE *FWA,LWA,LEN*  000620    DBG=IDP    4485    A
DBG=IDP   *        SYNTAX, AND IS USED MERELY AS A SPACE OPTIMIZATION BECAUSE  000630    DBG=IDP    4486    A
DBG=IDP   *        THIS SYNTAX NEEDS TO BE PARSED FOR A NUMBER OF DIFFERENT    000640    DBG=IDP    4487    A
DBG=IDP   *        STATEMENTS.                                         000650    DBG=IDP    4488    A
DBG=IDP   *                                                            000660    DBG=IDP    4489    A
DBG=IDP   *        ANY OR ALL OF THE SUBEXPRESSIONS WHICH CONSTITUTE THE       000670    DBG=IDP    4490    A
DBG=IDP   *        *FWA,LWA,LEN* TRIPLE MAY BE NULL.  IF A NULL SUBEXPRESSION  000680    DBG=IDP    4491    A
DBG=IDP   *        IS ENCOUNTERED, *PAT* WILL USE A DEFAULT VALUE THAT HAS     000690    DBG=IDP    4492    A
DBG=IDP   *        BEEN SET UP BY THE CALLER.                          000700    DBG=IDP    4493    A
DBG=IDP   *                                                            000710    DBG=IDP    4494    A
DBG=IDP   *        NOTE THAT *PAT* RESOLVES ANY POSSIBLE SYNTACTIC DIFFERENCES 000720    DBG=IDP    4495    A
DBG=IDP   *        BETWEEN THE USES OF THE *LWA* AND *LEN* PARAMETERS --       000730    DBG=IDP    4496    A
DBG=IDP   *                                                            000740    DBG=IDP    4497    A
DBG=IDP   *          1. FOR *FWA,LWA* (I.E. WHEN *LEN* IS NULL),       000750    DBG=IDP    4498    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *              NEED TO SET: LEN = LWA-FWA+1                    000760  DBG=IDP   4499   A
DBG=IDP  *                                                              000770  DBG=IDP   4500   A
DBG=IDP  *           2. FOR *FWA,,LEN* (I.E. WHEN *LWA* IS NULL),       000780  DBG=IDP   4501   A
DBG=IDP  *              NEED TO SET: LWA = FWA+LEN-1                    000790  DBG=IDP   4502   A
DBG=IDP  *                                                              000800  DBG=IDP   4503   A
DBG=IDP  *         THIS HOCUS POCUS IS NECESSARY TO INSURE THAT *LWA* AND *LEN* 000810  DBG=IDP   4504   A
DBG=IDP  *         DO NOT CONFLICT WITH EACH OTHER.                     000820  DBG=IDP   4505   A
DBG=IDP  *                                                              000830  DBG=IDP   4506   A
DBG=IDP  *         CONSIDER --                                          000840  DBG=IDP   4507   A
DBG=IDP  *                                                              000850  DBG=IDP   4508   A
DBG=IDP  *               SNAP 100,,2                                    000860  DBG=IDP   4509   A
DBG=IDP  *                                                              000870  DBG=IDP   4510   A
DBG=IDP  *         FOLLOWED BY --                                       000880  DBG=IDP   4511   A
DBG=IDP  *                                                              000890  DBG=IDP   4512   A
DBG=IDP  *               SNAP 100,110                                   000900  DBG=IDP   4513   A
DBG=IDP  *                                                              000910  DBG=IDP   4514   A
DBG=IDP  *         IF THE ABOVE *LWA/LEN* FIDDLING WERE NOT PERFORMED, UPON 000920  DBG=IDP   4515   A
DBG=IDP  *         ENCOUNTERING THE 2ND STMT, IDP WOULD NOT KNOW WHICH VALUE 000930  DBG=IDP   4516   A
DBG=IDP  *         TO USE: LEN=2 (RESIDUAL), OR LWA=110 (EXPLICIT).     000940  DBG=IDP   4517   A
DBG=IDP  *                                                              000950  DBG=IDP   4518   A
DBG=IDP  *         ENTRY  (A5+1)     = ADDR OF 1ST TOKEN IN TRIPLE, I.E. *FWA*. 000960  DBG=IDP   4519   A
DBG=IDP  *                (PAT#FWA) = DEFAULT *FWA*                     000970  DBG=IDP   4520   A
DBG=IDP  *                (PAT#LWA) = DEFAULT *LWA*                     000980  DBG=IDP   4521   A
DBG=IDP  *                (PAT#LEN) = DEFAULT *LEN*                     000990  DBG=IDP   4522   A
DBG=IDP  *                                                              001000  DBG=IDP   4523   A
DBG=IDP  *         EXIT   (A5,X5)    = A+C OF TOKEN THAT TERMINATED TRIPLE 001010  DBG=IDP   4524   A
DBG=IDP  *                (PAT#FWA) = FWA, AS SPECIFIED EXPLICITLY OR BY DEFAULT 001020  DBG=IDP   4525   A
DBG=IDP  *                (PAT#LWA) = LWA, AS SPECIFIED EXPLICITLY OR BY DEFAULT 001030  DBG=IDP   4526   A
DBG=IDP  *                (PAT#LEN) = LEN, AS SPECIFIED EXPLICITLY OR BY DEFAULT 001040  DBG=IDP   4527   A
DBG=IDP  *                (AP=FWA)  = (PAT#FWA) = FWA                   001050  DBG=IDP   4528   A
DBG=IDP  *                (AP=LWA)  = LWA, AS COMPUTED FROM *LWA* AND/OR *LEN* 001060  DBG=IDP   4529   A
DBG=IDP  *                (B2)      = .MI. IF FWA, LWA, OR LEN ARE BAD, ELSE .PL. 001070  DBG=IDP   4530   A
DBG=IDP  *                                                              001080  DBG=IDP   4531   A
DBG=IDP  *         USES   ALL                                           001090  DBG=IDP   4532   A
DBG=IDP  *                                                              001100  DBG=IDP   4533   A
DBG=IDP  *         CALLS  FLL,PAS                                       001110  DBG=IDP   4534   A
DBG=IDP                                                                 001120  DBG=IDP   4535   A
DBG=IDP                                                                 001130  DBG=IDP   4536   A
DBG=IDP   PAT      SUBR              ** ENTRY/EXIT **                   001140  DBG=IDP   4537   A
DBG=IDP                                                                 001150  DBG=IDP   4538   A
DBG=IDP  *         INITIALIZATION.                                      001160  DBG=IDP   4539   A
DBG=IDP                                                                 001170  DBG=IDP   4540   A
DBG=IDP            BX6    X6-X6                                         001180  DBG=IDP   4541   A
DBG=IDP            SA6    PATFLG      CLEAR *PATFLG*                    001190  DBG=IDP   4542   A
DBG=IDP                                                                 001200  DBG=IDP   4543   A
DBG=IDP  *         PARSE/PROCESS FWA.                                   001210  DBG=IDP   4544   A
DBG=IDP                                                                 001220  DBG=IDP   4545   A
DBG=IDP            ZR     X5,PAT4     IF EOS ENCOUNTERED                001230  DBG=IDP   4546   A
DBG=IDP            RJ     PAS         PARSE SUBEXPRESSION -- FWA        001240  DBG=IDP   4547   A
DBG=IDP            ZR     X3,PAT2     IF FWA IS NULL                    001250  DBG=IDP   4548   A
DBG=IDP            SX6    X1+                                           001260  DBG=IDP   4549   A
DBG=IDP            SA6    PAT#FWA                                       001270  DBG=IDP   4550   A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP                                                                  001280   DBG=IDP     4551     A
DBG=IDP  *         PARSE/PROCESS LWA.                                    001290   DBG=IDP     4552     A
DBG=IDP                                                                  001300   DBG=IDP     4553     A
DBG=IDP   PAT2     ZR     X5,PAT4     IF EOS ENCOUNTERED                 001310   DBG=IDP     4554     A
DBG=IDP            RJ     PAS         PARSE SUBEXPRESSION -- LWA         001320   DBG=IDP     4555     A
DBG=IDP            ZR     X3,PAT3     IF LWA IS NULL                     001330   DBG=IDP     4556     A
DBG=IDP            SX6    X1                                             001340   DBG=IDP     4557     A
DBG=IDP            SX7    PAF.LWAM                                       001350   DBG=IDP     4558     A
DBG=IDP            SA6    PAT#LWA                                        001360   DBG=IDP     4559     A
DBG=IDP            SA7    PATFLG                                         001370   DBG=IDP     4560     A
DBG=IDP                                                                  001380   DBG=IDP     4561     A
DBG=IDP  *         PARSE/PROCESS LEN.                                    001390   DBG=IDP     4562     A
DBG=IDP                                                                  001400   DBG=IDP     4563     A
DBG=IDP   PAT3     ZR     X5,PAT4     IF EOS ENCOUNTERED                 001410   DBG=IDP     4564     A
DBG=IDP            RJ     PAS         PARSE SUBEXPRESSION -- LEN         001420   DBG=IDP     4565     A
DBG=IDP            ZR     X3,PAT4     IF LEN IS NULL                     001430   DBG=IDP     4566     A
DBG=IDP            SA2    PATFLG                                         001440   DBG=IDP     4567     A
DBG=IDP            SX7    PAF.LENM                                       001450   DBG=IDP     4568     A
DBG=IDP            SX6    X1                                             001460   DBG=IDP     4569     A
DBG=IDP            BX7    X2+X7       SET TO *LEN SPECIFIED*             001470   DBG=IDP     4570     A
DBG=IDP            SA6    PAT#LEN                                        001480   DBG=IDP     4571     A
DBG=IDP            SA7    A2+                                            001490   DBG=IDP     4572     A
DBG=IDP                                                                  001500   DBG=IDP     4573     A
DBG=IDP  *         HERE TO RESOLVE ANY POSSIBLE SYNTACTIC DIFFERENCES    001510   DBG=IDP     4574     A
DBG=IDP  *         BETWEEN *FWA,LWA* AND *FWA,,LEN*.                     001520   DBG=IDP     4575     A
DBG=IDP                                                                  001530   DBG=IDP     4576     A
DBG=IDP   PAT4     SA1    PATFLG                                         001540   DBG=IDP     4577     A
DBG=IDP            SA2    PAT#FWA     (X2) = FWA                         001550   DBG=IDP     4578     A
DBG=IDP            SA3    A2+B1       (X3) = LWA                         001560   DBG=IDP     4579     A
DBG=IDP            SA4    A3+B1       (X4) = LEN                         001570   DBG=IDP     4580     A
DBG=IDP            ZR     X1,PAT6     IF NO PROBLEM                      001580   DBG=IDP     4581     A
DBG=IDP            SX6    X1-PAF.LWAM-PAF.LENM                           001590   DBG=IDP     4582     A
DBG=IDP            SX7    X1-PAF.LWAM                                    001600   DBG=IDP     4583     A
DBG=IDP            ZR     X6,PAT6     IF *FWA,LWA,LEN*                   001610   DBG=IDP     4584     A
DBG=IDP            ZR     X7,PAT5     IF *FWA,LWA*                       001620   DBG=IDP     4585     A
DBG=IDP                                                                  001630   DBG=IDP     4586     A
DBG=IDP  *         HERE IF *FWA,,LEN*.                                   001640   DBG=IDP     4587     A
DBG=IDP  *         SET: LWA = FWA+LEN-1                                  001650   DBG=IDP     4588     A
DBG=IDP                                                                  001660   DBG=IDP     4589     A
DBG=IDP            IX6    X2+X4                                          001670   DBG=IDP     4590     A
DBG=IDP            SX3    X6-1                                           001680   DBG=IDP     4591     A
DBG=IDP            BX6    X3                                             001690   DBG=IDP     4592     A
DBG=IDP            SA6    A3                                             001700   DBG=IDP     4593     A
DBG=IDP            EQ     PAT6                                           001710   DBG=IDP     4594     A
DBG=IDP                                                                  001720   DBG=IDP     4595     A
DBG=IDP  *         HERE IF *FWA,LWA*.                                    001730   DBG=IDP     4596     A
DBG=IDP  *         SET: LEN = LWA-FWA+1                                  001740   DBG=IDP     4597     A
DBG=IDP                                                                  001750   DBG=IDP     4598     A
DBG=IDP   PAT5     IX6    X3-X2                                          001760   DBG=IDP     4599     A
DBG=IDP            SX4    X6+B1                                          001770   DBG=IDP     4600     A
DBG=IDP            BX6    X4                                             001780   DBG=IDP     4601     A
DBG=IDP            SA6    A4                                             001790   DBG=IDP     4602     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP                                                                  001800   DBG=IDP     4603     A
DBG=IDP  *         CHECK *FWA,LWA,LEN*.                                  001810   DBG=IDP     4604     A
DBG=IDP                                                                  001820   DBG=IDP     4605     A
DBG=IDP   PAT6     RJ     FLL         CHECK *FWA,LWA,LEN*               001830   DBG=IDP     4606     A
DBG=IDP            PL     B2,EXIT.    IF FWA, LWA, AND LEN OK           001840   DBG=IDP     4607     A
DBG=IDP            BX6    X6-X6                                         001850   DBG=IDP     4608     A
DBG=IDP            SA6    PAT#FWA     CLEAR *FWA*                       001860   DBG=IDP     4609     A
DBG=IDP            SA6    A6+B1       CLEAR *LWA*                       001870   DBG=IDP     4610     A
DBG=IDP            SA6    A6+B1       CLEAR *LEN*                       001880   DBG=IDP     4611     A
DBG=IDP            EQ     EXIT.                                         001890   DBG=IDP     4612     A
DBG=IDP                                                                  001900   DBG=IDP     4613     A
DBG=IDP                                                                  001910   DBG=IDP     4614     A
DBG=IDP            LOC    AP=LL                                         001920   DBG=IDP     4615     A
DBG=IDP   PAT#FWA  =      *                                             001930   DBG=IDP     4616     A
DBG=IDP   PAT#LWA  =      *+1                                           001940   DBG=IDP     4617     A
DBG=IDP   PAT#LEN  =      *+2                                           001950   DBG=IDP     4618     A
DBG=IDP            LOC    *O                                            001960   DBG=IDP     4619     A
DBG=IDP   POL      SPACE  4,8                                                     DBG=IDP     4620     A
DBG=IDP  **        POL - PROCESS OPTIONS LIST ITEM.                               DBG=IDP     4621     A
DBG=IDP  *                                                                        DBG=IDP     4622     A
DBG=IDP  *                                                                        DBG=IDP     4623     A
DBG=IDP  *                THIS ROUTINE WILL PROCESS AN OPTIONS LIST ITEM THAT IS  DBG=IDP     4624     A
DBG=IDP  *         TERMINATED BY A -,- OR -EOS-.                                  DBG=IDP     4625     A
DBG=IDP  *                                                                        DBG=IDP     4626     A
DBG=IDP  *         ENTRY  (B6) = FWA OF OPTIONS KEYWORD TABLE (OKT)               DBG=IDP     4627     A
DBG=IDP  *                (A5+1,X5) = ADDR + CONTENTS OF 1ST TOKEN IN LIST        DBG=IDP     4628     A
DBG=IDP  *                                                                        DBG=IDP     4629     A
DBG=IDP  *         EXIT   IF NO ERROR--                                           DBG=IDP     4630     A
DBG=IDP  *                (A5,X5) =  A + C OF TOKEN THAT TERMINATED OPTIONS LIST  DBG=IDP     4631     A
DBG=IDP  *                (X3)    = .ZR. IF OPTIONS LIST ITEM WAS NULL, ELSE .NZ. DBG=IDP     4632     A
DBG=IDP  *                (X6)    =  -DEFAULT IF - PREFIX, ELSE +DEFAULT          DBG=IDP     4633     A
DBG=IDP  *                (X7)    =  +0 IF - PREFIX, ELSE + DEFAULT               DBG=IDP     4634     A
DBG=IDP  *                (B7)    =  ORDINAL INTO *OKT* OF FIND                   DBG=IDP     4635     A
DBG=IDP  *                                                                        DBG=IDP     4636     A
DBG=IDP  *                IF AN ERROR-- EXIT IS TO *IDP=ER*                       DBG=IDP     4637     A
DBG=IDP  *                                                                        DBG=IDP     4638     A
DBG=IDP  *         USES   X - ALL                                                 DBG=IDP     4639     A
DBG=IDP  *                A - 2,3,5                                               DBG=IDP     4640     A
DBG=IDP  *                B - 7                                                   DBG=IDP     4641     A
DBG=IDP  *                                                                        DBG=IDP     4642     A
DBG=IDP  *         CALLS  IDP=ER,SKT                                              DBG=IDP     4643     A
DBG=IDP                                                                           DBG=IDP     4644     A
DBG=IDP                                                                           DBG=IDP     4645     A
DBG=IDP   POL      SUBR               ** ENTRY/EXIT **                            DBG=IDP     4646     A
DBG=IDP                                                                           DBG=IDP     4647     A
DBG=IDP  *         INITIALIZATION.                                                DBG=IDP     4648     A
DBG=IDP                                                                           DBG=IDP     4649     A
DBG=IDP            BX4    X4-X4       SET TO *NO - PREFIX SO FAR*                DBG=IDP     4650     A
DBG=IDP            MX6    -1                                                     DBG=IDP     4651     A
DBG=IDP            SX7    B0                                                     DBG=IDP     4652     A
DBG=IDP            SA6    IDPNUL      SET TO *NULL EXPRESSION SO FAR*            DBG=IDP     4653     A
DBG=IDP            SA7    IDPACC      CLEAR ACCUMULATOR                          DBG=IDP     4654     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP                                                                  DBG=IDP    4655    A
DBG=IDP  *         MAIN LOOP NODE.                                       DBG=IDP    4656    A
DBG=IDP                                                                  DBG=IDP    4657    A
DBG=IDP   POL2     SA3    IDPNUL                                         DBG=IDP    4658    A
DBG=IDP            SA5    A5+B1                                          DBG=IDP    4659    A
DBG=IDP            SX6    X3+B1                                          DBG=IDP    4660    A
DBG=IDP            MX0    -CHAR                                          DBG=IDP    4661    A
DBG=IDP            SA6    A3                                             DBG=IDP    4662    A
DBG=IDP            ZR     X5,POL4     IF EOS ENCOUNTERED                 DBG=IDP    4663    A
DBG=IDP            SX6    X5-O.SEP                                       DBG=IDP    4664    A
DBG=IDP            SX7    X5-O.VAR                                       DBG=IDP    4665    A
DBG=IDP            NZ     X6,POL3     IF NOT A SEPARATOR                 DBG=IDP    4666    A
DBG=IDP                                                                  DBG=IDP    4667    A
DBG=IDP  *         HERE IF A SEPARATOR TOKEN ENCOUNTERED.                DBG=IDP    4668    A
DBG=IDP                                                                  DBG=IDP    4669    A
DBG=IDP            LX5    1*CHAR-10*CHAR                                 DBG=IDP    4670    A
DBG=IDP            BX5    -X0*X5                                         DBG=IDP    4671    A
DBG=IDP            SX6    X5-1R,                                         DBG=IDP    4672    A
DBG=IDP            SX7    X5-1R-                                         DBG=IDP    4673    A
DBG=IDP            ZR     X6,POL4     IF , ENCOUNTERED (I.E. END OF LIST) DBG=IDP   4674    A
DBG=IDP            NZ     X7,IDP=ER   IF NOT - TOKEN, ERROR...           DBG=IDP    4675    A
DBG=IDP            MX4    60          SET TO * - PREFIX ENCOUNTERED*     DBG=IDP    4676    A
DBG=IDP            EQ     POL2                                           DBG=IDP    4677    A
DBG=IDP                                                                  DBG=IDP    4678    A
DBG=IDP  *         HERE TO PROCESS NAME TOKEN.                           DBG=IDP    4679    A
DBG=IDP                                                                  DBG=IDP    4680    A
DBG=IDP   POL3     NZ     X7,IDP=ER   IF NOT A NAME TOKEN, ERROR...      DBG=IDP    4681    A
DBG=IDP            MX0    7*CHAR                                         DBG=IDP    4682    A
DBG=IDP            SX6    B1                                             DBG=IDP    4683    A
DBG=IDP            SA2    =40404040404040404040B                         DBG=IDP    4684    A
DBG=IDP            BX1    X0*X5       (X1) = 0LKEYWORD                   DBG=IDP    4685    A
DBG=IDP            IX6    X1-X6       BORROW RIPPLES LEFT TO 1ST NON-ZERO BIT DBG=IDP 4686 A
DBG=IDP            BX7    -X6+X1      (X7) = ALL TRAILING 0 BITS IN (X1) = 0 BITS DBG=IDP 4687 A
DBG=IDP  *                                ALL OTHER LEADING BITS = 1 BITS  DBG=IDP    4688    A
DBG=IDP            SB7    60-5        (B7) = RIGHT CIRCULAR SHIFT 5 BITS DBG=IDP    4689    A
DBG=IDP            BX6    X2*X7       (X6) = 40B WHERE EACH CHAR IN KEYWORD IS DBG=IDP 4690 A
DBG=IDP            LX7    X6,B7       (X7) = 01B WHERE EACH CHAR IN KEYWORD IS DBG=IDP 4691 A
DBG=IDP            IX7    X6-X7       (X7) = 37B WHERE EACH CHAR IN KEYWORD IS DBG=IDP 4692 A
DBG=IDP            BX0    X6+X7       (X0) = 77B WHERE EACH CHAR IN KEYWORD IS DBG=IDP 4693 A
DBG=IDP            RJ     SKT         SEARCH FOR NAME                    DBG=IDP    4694    A
DBG=IDP            ZR     X2,IDP=ER   IF NO FIND                         DBG=IDP    4695    A
DBG=IDP            SX6    X2                                             DBG=IDP    4696    A
DBG=IDP            SA6    IDPACC                                         DBG=IDP    4697    A
DBG=IDP            EQ     POL2                                           DBG=IDP    4698    A
DBG=IDP                                                                  DBG=IDP    4699    A
DBG=IDP  *         HERE TO SET UP EXIT CONDITIONS.                       DBG=IDP    4700    A
DBG=IDP                                                                  DBG=IDP    4701    A
DBG=IDP   POL4     SA1    IDPACC                                         DBG=IDP    4702    A
DBG=IDP            SA3    IDPNUL                                         DBG=IDP    4703    A
DBG=IDP            BX6    X1-X4       (X6) = -DEFAULT IF - PREFIX, ELSE +DEFAULT DBG=IDP 4704 A
DBG=IDP            BX7    -X4*X1      (X7) = +0 IF - PREFIX, ELSE +DEFAULT DBG=IDP 4705 A
DBG=IDP            EQ     EXIT.                                          DBG=IDP    4706    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP   RIL      SPACE  4,8                                                  DBG=IDP     4707     A
DBG=IDP  **        RIL - READ IDP INPUT LINE.                                  DBG=IDP     4708     A
DBG=IDP  *                                                                     DBG=IDP     4709     A
DBG=IDP  *                                                                     DBG=IDP     4710     A
DBG=IDP  *         ENTRY  IF READING INTERACTIVELY (F.IDI) -- NONE             DBG=IDP     4711     A
DBG=IDP  *                IF READING FROM BATCH (F.BDI) -- INITIAL *READ* ISSUED DBG=IDP  4712     A
DBG=IDP  *                  TO FILL BUFFER                                     DBG=IDP     4713     A
DBG=IDP  *                                                                     DBG=IDP     4714     A
DBG=IDP  *         EXIT   (X1)      = .NZ. IF EOR/EOF ENCOUNTERED, ELSE .ZR.  DBG=IDP     4715     A
DBG=IDP  *                IF NO EOR/EOF ENCOUNTERED --                         DBG=IDP     4716     A
DBG=IDP  *                (A5,X5)   = A+C OF 1ST WORD OF LINE IMAGE            DBG=IDP     4717     A
DBG=IDP  *                (L=LIM)   = NR OF WORDS IN SOURCE LINE IMAGE         DBG=IDP     4718     A
DBG=IDP  *                              (INCLUDES WORD CONTAINING EOL MARK)    DBG=IDP     4719     A
DBG=IDP  *                                                                     DBG=IDP     4720     A
DBG=IDP  *         USES   ALL BUT A0,X0     (INCLUDES ALL CALLS)              DBG=IDP     4721     A
DBG=IDP  *                                                                     DBG=IDP     4722     A
DBG=IDP  *         CALLS  READ,READC                                           DBG=IDP     4723     A
DBG=IDP                                                                        DBG=IDP     4724     A
DBG=IDP                                                                        DBG=IDP     4725     A
DBG=IDP   RIL      SUBR               ** ENTRY/EXIT **                         DBG=IDP     4726     A
DBG=IDP                                                                        DBG=IDP     4727     A
DBG=IDP   RIL2     SA5    IDPFLG                                               DBG=IDP     4728     A
DBG=IDP            SX2    F.IDI                                                DBG=IDP     4729     A
DBG=IDP            LX5    59-IDF.INPP                                          DBG=IDP     4730     A
DBG=IDP            SB6    IDPLIM                                               DBG=IDP     4731     A
DBG=IDP            PL     X5,RIL3     IF READING INTERACTIVELY (F.IDI)         DBG=IDP     4732     A
DBG=IDP            SX2    F.BDI                                                DBG=IDP     4733     A
DBG=IDP                                                                        DBG=IDP     4734     A
DBG=IDP   RIL3     READC  X2,B6,8                                              DBG=IDP     4735     A
DBG=IDP            ZR     X1,RIL4     IF NOT EOR/EOF STATUS                    DBG=IDP     4736     A
DBG=IDP            MI     X5,EXIT.    IF READING FROM BATCH (F.BDI)            DBG=IDP     4737     A
DBG=IDP            READ   X2,RCL                                               DBG=IDP     4738     A
DBG=IDP                                                                        DBG=IDP     4739     A
DBG=IDP  *         SET EOR STATUS TO PREVENT *READC* FROM READING AHEAD.       DBG=IDP     4740     A
DBG=IDP                                                                        DBG=IDP     4741     A
DBG=IDP            SA1    X2                                                   DBG=IDP     4742     A
DBG=IDP            MX6    60-18                                                DBG=IDP     4743     A
DBG=IDP            BX1    X6*X1                                                DBG=IDP     4744     A
DBG=IDP            SX6    040021B                                              DBG=IDP     4745     A
DBG=IDP            BX6    X1+X6                                                DBG=IDP     4746     A
DBG=IDP            SA6    A1                                                   DBG=IDP     4747     A
DBG=IDP            EQ     RIL2                                                 DBG=IDP     4748     A
DBG=IDP                                                                        DBG=IDP     4749     A
DBG=IDP  *         SET UP EXIT CONDITIONS.                                     DBG=IDP     4750     A
DBG=IDP                                                                        DBG=IDP     4751     A
DBG=IDP   RIL4     SX7    B6-IDPLIM   (X7) = NR OF WORDS IN LINE IMAGE         DBG=IDP     4752     A
DBG=IDP            SA5    IDPLIM      (A5,X5) = A+C OF 1ST WORD OF LINE IMAGE  DBG=IDP     4753     A
DBG=IDP            SA7    L=LIM                                                DBG=IDP     4754     A
DBG=IDP            EQ     EXIT.                                                DBG=IDP     4755     A
DBG=IDP   SKT      SPACE  4,8                                                  DBG=IDP     4756     A
DBG=IDP  **        SKT - SEARCH KEYWORD TABLE.                                 DBG=IDP     4757     A
DBG=IDP  *                                                                     DBG=IDP     4758     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *                                                                  DBG=IDP    4759    A
DBG=IDP  *                SEARCHS A TABLE THAT IS TERMINATED BY A ZERO WORD  DBG=IDP    4760    A
DBG=IDP  *         FOR A MATCH, USING A SPECIFIED EXTRACTION MASK.          DBG=IDP    4761    A
DBG=IDP  *                                                                  DBG=IDP    4762    A
DBG=IDP  *         ENTRY  (B6)     =  FWA TO BEGIN SEARCH                   DBG=IDP    4763    A
DBG=IDP  *                (X0)     =  EXTRACTION MASK TO USE                DBG=IDP    4764    A
DBG=IDP  *                (X1)     =  ELEMENT TO SEARCH FOR                 DBG=IDP    4765    A
DBG=IDP  *                                                                  DBG=IDP    4766    A
DBG=IDP  *         EXIT   IF NO FIND --                                     DBG=IDP    4767    A
DBG=IDP  *                (X2)     =  .ZR.                                  DBG=IDP    4768    A
DBG=IDP  *                IF A FIND --                                     DBG=IDP    4769    A
DBG=IDP  *                (B7)     =  ORDINAL OF FIND                       DBG=IDP    4770    A
DBG=IDP  *                (A2,X2)  =  ADDR + CONTENTS OF WORD THAT MATCHED  DBG=IDP    4771    A
DBG=IDP  *                                                                  DBG=IDP    4772    A
DBG=IDP  *         USES   X - 2,3,6,7                                       DBG=IDP    4773    A
DBG=IDP  *                A - 2                                            DBG=IDP    4774    A
DBG=IDP  *                B - 7                                            DBG=IDP    4775    A
DBG=IDP  *                                                                  DBG=IDP    4776    A
DBG=IDP  *         CALLS  NONE                                              DBG=IDP    4777    A
DBG=IDP                                                                     DBG=IDP    4778    A
DBG=IDP                                                                     DBG=IDP    4779    A
DBG=IDP   SKT      SUBR              ** ENTRY/EXIT **                       DBG=IDP    4780    A
DBG=IDP            SA2    B6                                                DBG=IDP    4781    A
DBG=IDP            BX6    X0*X1                                             DBG=IDP    4782    A
DBG=IDP            SB7    -1                                                DBG=IDP    4783    A
DBG=IDP                                                                     DBG=IDP    4784    A
DBG=IDP   SKT2     ZR     X2,EXIT.    IF END OF TABLE                       DBG=IDP    4785    A
DBG=IDP            BX7    X0*X2                                             DBG=IDP    4786    A
DBG=IDP            SB7    B7+B1                                             DBG=IDP    4787    A
DBG=IDP            IX3    X6-X7                                             DBG=IDP    4788    A
DBG=IDP            ZR     X3,EXIT.    IF A FIND                             DBG=IDP    4789    A
DBG=IDP            SA2    A2+B1                                             DBG=IDP    4790    A
DBG=IDP            EQ     SKT2                                              DBG=IDP    4791    A
DBG=IDP   STP      SPACE  4,8                                               DBG=IDP    4792    A
DBG=IDP  **        STP - STEP AN INSTRUCTION.                               DBG=IDP    4793    A
DBG=IDP  *                                                                  DBG=IDP    4794    A
DBG=IDP  *                                                                  DBG=IDP    4795    A
DBG=IDP  *                THIS ROUTINE WILL STEP A SINGLE CPU INSTRUCTION, AND  DBG=IDP    4796    A
DBG=IDP  *         LIST THE RESULT REGISTER, IF APPROPRIATE.                DBG=IDP    4797    A
DBG=IDP  *                                                                  DBG=IDP    4798    A
DBG=IDP  *                *STP* ALSO PERFORMS A LIMITED SET OF ERROR CHECKS FOR  DBG=IDP    4799    A
DBG=IDP  *         RANGE ERRORS (MODE=1). 1) THE PSEUDO P REGISTER IS CHECKED,  DBG=IDP    4800    A
DBG=IDP  *         2) XFERS OF CONTROL (BRANCH OR RJ) ARE CHECKED, AND      DBG=IDP    4801    A
DBG=IDP  *         3) LOAD/STORE ADDRESSES ARE CHECKED. IF AN ERROR IS DETECTED,  DBG=IDP    4802    A
DBG=IDP  *         THE BAD INSTRUCTION AND THE -P- REGISTER ARE DISPLAYED (EVEN  DBG=IDP    4803    A
DBG=IDP  *         IF -L SELECTED). STEP MODE IS THEN TERMINATED.           DBG=IDP    4804    A
DBG=IDP  *                                                                  DBG=IDP    4805    A
DBG=IDP  *         ENTRY  (ST=ENDX) = CONTAINS THE PREVIOUS CONTENTS OF IDP  DBG=IDP    4806    A
DBG=IDP  *                            GENERATED BREAK ADDR IF THEY NEED TO  DBG=IDP    4807    A
DBG=IDP  *                            BE XEQ, ELSE .ZR.                     DBG=IDP    4808    A
DBG=IDP  *                (IDPPREG) = PSEUDO P REGISTER, I.E. ADDRESS OF WORD  DBG=IDP    4809    A
DBG=IDP  *                            CONTAINING INSTRUCTION TO XEQ         DBG=IDP    4810    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *                (IDPPOS)  = POSITION COUNTER. LEFT MOST BIT OF INSTR          DBG=IDP    4811    A
DBG=IDP  *                            TO XEQ IS BIT (IDPPOS)-1 IN ((IDPPREG)).          DBG=IDP    4812    A
DBG=IDP  *                            (BITS ARE NUMBERED 59 THRU 0)                     DBG=IDP    4813    A
DBG=IDP  *                            E.G. FOR A WORD CONTAINING--                      DBG=IDP    4814    A
DBG=IDP  *                             +  MX0  -CHAR                                    DBG=IDP    4815    A
DBG=IDP  *                                SA1  ADDR                                     DBG=IDP    4816    A
DBG=IDP  *                                BX6  -X0*X5                                   DBG=IDP    4817    A
DBG=IDP  *                            WHEN STEPPING *SA1  ADDR*--                       DBG=IDP    4818    A
DBG=IDP  *                            (IDPPOS) = 45                                     DBG=IDP    4819    A
DBG=IDP  *                (IDPXLST) = .NZ. IF LISTING EACH STEPPED INSTRUCTION,         DBG=IDP    4820    A
DBG=IDP  *                                ELSE .ZR.                                     DBG=IDP    4821    A
DBG=IDP  *                                                                              DBG=IDP    4822    A
DBG=IDP  *         EXIT   (ST=ENDX), (IDPPREG), AND (IDPPOS) UPDATED                    DBG=IDP    4823    A
DBG=IDP  *                TO *IDP=ER* IF AN ERROR DETECTED                              DBG=IDP    4824    A
DBG=IDP  *                TO *IDP2A* IF AN *RJ IDP=* IS ENCOUNTERED AND THE             DBG=IDP    4825    A
DBG=IDP  *                  BREAK CONDITIONS ARE STATISFIED                             DBG=IDP    4826    A
DBG=IDP  *                                                                              DBG=IDP    4827    A
DBG=IDP  *         USES   ALL BUT A0                                                    DBG=IDP    4828    A
DBG=IDP  *                                                                              DBG=IDP    4829    A
DBG=IDP  *         CALLS  BRK,CHK,DAB,DAZ,DUX,FRA=(IF DEF),FRK,                         DBG=IDP    4830    A
DBG=IDP  *                  PRIDP,RSR=,SVR=                                             DBG=IDP    4831    A
DBG=IDP                                                                                 DBG=IDP    4832    A
DBG=IDP                                                                                 DBG=IDP    4833    A
DBG=IDP   STP      SUBR                ** ENTRY/EXIT **                                 DBG=IDP    4834    A
DBG=IDP                                                                                 DBG=IDP    4835    A
DBG=IDP  *         INITIALIZATION.                                                      DBG=IDP    4836    A
DBG=IDP                                                                                 DBG=IDP    4837    A
DBG=IDP            MX6    0                                                             DBG=IDP    4838    A
DBG=IDP            SX7    -B1                                                           DBG=IDP    4839    A
DBG=IDP            SA6    IDPXFR      CLEAR *TRANSFER CONTROL* FLAG                     DBG=IDP    4840    A
DBG=IDP            SA7    IDPXAR      SET TO *NO ADDR REF SO FAR*                       DBG=IDP    4841    A
DBG=IDP                                                                                 DBG=IDP    4842    A
DBG=IDP  *         FETCH WORD CONTAINING INSTRUCTION TO BE XEQ.                         DBG=IDP    4843    A
DBG=IDP                                                                                 DBG=IDP    4844    A
DBG=IDP            SA1    IDPPREG     (X1) = PSEUDO P REGISTER                          DBG=IDP    4845    A
DBG=IDP            SA2    ST=ENDX                                                       DBG=IDP    4846    A
DBG=IDP            SA3    A1+B1       (X3) = POS COUNTER                                DBG=IDP    4847    A
DBG=IDP            SA5    X1          (X5) = WORD CONTAINING INSTRUCTION TO XEQ         DBG=IDP    4848    A
DBG=IDP            ZR     X2,STP2     IF NOT *IDP* GENERATED BREAK                      DBG=IDP    4849    A
DBG=IDP                                                                                 DBG=IDP    4850    A
DBG=IDP  *         HERE BECAUSE INSTRUCTION TO XEQ AT *ST=ENDX*.                        DBG=IDP    4851    A
DBG=IDP                                                                                 DBG=IDP    4852    A
DBG=IDP            SA5    A5-B1       (A5) = ADDR OF GENERATED BREAK                    DBG=IDP    4853    A
DBG=IDP            BX5    X2                                                            DBG=IDP    4854    A
DBG=IDP                                                                                 DBG=IDP    4855    A
DBG=IDP  *         LIST INSTRUCTION IF NECESSARY.                                       DBG=IDP    4856    A
DBG=IDP                                                                                 DBG=IDP    4857    A
DBG=IDP   STP2     SA1    IDPXLST                                                       DBG=IDP    4858    A
DBG=IDP            SB4    X3                                                            DBG=IDP    4859    A
DBG=IDP            ZR     X1,STP3     IF NOT LISTING INSTRUCTION                        DBG=IDP    4860    A
DBG=IDP            RJ     DAZ         DEASSEMBLE INSTRUCTION                            DBG=IDP    4861    A
DBG=IDP                                                                                 DBG=IDP    4862    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *        BREAK OUT OPCODE.                                          DBG=IDP    4863    A
DBG=IDP                                                                      DBG=IDP    4864    A
DBG=IDP   STP3    SB6    -B4                                                 DBG=IDP    4865    A
DBG=IDP           MX0    -6                                                  DBG=IDP    4866    A
DBG=IDP           SB6    B6+60+6                                             DBG=IDP    4867    A
DBG=IDP           LX5    X5,B6       (X5) = 54/OTHER STUFF, 6/OPCODE         DBG=IDP    4868    A
DBG=IDP           BX6    -X0*X5                                              DBG=IDP    4869    A
DBG=IDP           LX5    3           (X5) = 51/OTHER, 6/OPC, 3/I             DBG=IDP    4870    A
DBG=IDP           MX0    -3                                                  DBG=IDP    4871    A
DBG=IDP           SA6    STP=OP                                              DBG=IDP    4872    A
DBG=IDP           ZR     X6,STP=ER   IF 00B OPCODE, TRYING TO XEQ DATA...    DBG=IDP    4873    A
DBG=IDP                                                                      DBG=IDP    4874    A
DBG=IDP  *        BREAK OUT *I* FIELD.                                       DBG=IDP    4875    A
DBG=IDP                                                                      DBG=IDP    4876    A
DBG=IDP           BX6    -X0*X5                                              DBG=IDP    4877    A
DBG=IDP           LX5    3           (X5) = 48/OTHER, 6/OPC, 3/I, 3/J        DBG=IDP    4878    A
DBG=IDP           SA6    A6+B1                                               DBG=IDP    4879    A
DBG=IDP                                                                      DBG=IDP    4880    A
DBG=IDP  *        BREAK OUT *J* FIELD.                                       DBG=IDP    4881    A
DBG=IDP                                                                      DBG=IDP    4882    A
DBG=IDP           BX6    -X0*X5                                              DBG=IDP    4883    A
DBG=IDP           LX5    3           (X5) = 45/OTHER, 6/OPC, 3/I, 3/J, 3/K   DBG=IDP    4884    A
DBG=IDP           SA6    A6+B1                                               DBG=IDP    4885    A
DBG=IDP                                                                      DBG=IDP    4886    A
DBG=IDP  *        BREAK OUT *K* FIELD.                                       DBG=IDP    4887    A
DBG=IDP                                                                      DBG=IDP    4888    A
DBG=IDP           BX6    -X0*X5                                              DBG=IDP    4889    A
DBG=IDP           LX5    18-3        (X5) = 30/OTHER, 6/OPC, 3/I, 3/J, 18/Q  DBG=IDP    4890    A
DBG=IDP           SA6    A6+B1                                               DBG=IDP    4891    A
DBG=IDP                                                                      DBG=IDP    4892    A
DBG=IDP  *        BREAK OUT *Q* 18 BIT ADDRESS FIELD.                        DBG=IDP    4893    A
DBG=IDP                                                                      DBG=IDP    4894    A
DBG=IDP           SX6    X5                                                  DBG=IDP    4895    A
DBG=IDP           SA6    STP=ADR                                             DBG=IDP    4896    A
DBG=IDP                                                                      DBG=IDP    4897    A
DBG=IDP  *        FETCH INSTRUCTION SKELETON AND CHECK FOR LENGTH CONFLICT.  DBG=IDP    4898    A
DBG=IDP                                                                      DBG=IDP    4899    A
DBG=IDP           SA1    STP=OP                                              DBG=IDP    4900    A
DBG=IDP           LX5    30          LEFT JUSTIFY INSTRUCTION                DBG=IDP    4901    A
DBG=IDP           MX0    15                                                  DBG=IDP    4902    A
DBG=IDP           SA2    DAZ=PS+X1   (X2) = INSTRUCTION SKELETON             DBG=IDP    4903    A
DBG=IDP           SB5    15                                                  DBG=IDP    4904    A
DBG=IDP           PL     X2,STP4     IF SHORT (15 BIT) INSTRUCTION           DBG=IDP    4905    A
DBG=IDP           MX0    30                                                  DBG=IDP    4906    A
DBG=IDP           SB5    B5+B5                                               DBG=IDP    4907    A
DBG=IDP                                                                      DBG=IDP    4908    A
DBG=IDP   STP4    LT     B4,B5,STP=ER IF NOT ENOUGH ROOM FOR 30 BIT INSTRUCTION,  DBG=IDP    4909    A
DBG=IDP  *                            TRYING TO XEQ DATA...                  DBG=IDP    4910    A
DBG=IDP                                                                      DBG=IDP    4911    A
DBG=IDP  *        BREAK OUT FULL INSTRUCTION.                                DBG=IDP    4912    A
DBG=IDP                                                                      DBG=IDP    4913    A
DBG=IDP           SA3    =46000460004600046000B                              DBG=IDP    4914    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            BX6    X0*X5                                                  DBG=IDP    4915    A
DBG=IDP            BX7    -X0*X3                                                 DBG=IDP    4916    A
DBG=IDP            IX6    X6+X7       (X6) = INSTRUCTION, LEFT JUST W/ NO-OP FILL DBG=IDP    4917    A
DBG=IDP            SA6    STP=IN                                                 DBG=IDP    4918    A
DBG=IDP                                                                           DBG=IDP    4919    A
DBG=IDP  *         SPLIT INSTRUCTION INTO REPRESENTATIVE TYPE--                   DBG=IDP    4920    A
DBG=IDP  *           01 THRU 07  BRANCH                                           DBG=IDP    4921    A
DBG=IDP  *           10 THRU 47  NORMAL                                           DBG=IDP    4922    A
DBG=IDP  *           50 THRU 57  LOAD/STORE                                       DBG=IDP    4923    A
DBG=IDP  *           60 THRU 77  NORMAL                                           DBG=IDP    4924    A
DBG=IDP                                                                           DBG=IDP    4925    A
DBG=IDP            SX3    X1-10B                                                 DBG=IDP    4926    A
DBG=IDP            SX4    X1-50B                                                 DBG=IDP    4927    A
DBG=IDP            MI     X3,STP6     IF A BRANCH                                DBG=IDP    4928    A
DBG=IDP            MI     X4,STP5     IF A NORMAL                                DBG=IDP    4929    A
DBG=IDP            SX3    X1-60B                                                 DBG=IDP    4930    A
DBG=IDP            MI     X3,STP12    IF A LOAD/STORE                            DBG=IDP    4931    A
DBG=IDP                                                                           DBG=IDP    4932    A
DBG=IDP  *         HERE IF A NORMAL INSTRUCTION (NO SPECIAL PROCESSING).          DBG=IDP    4933    A
DBG=IDP                                                                           DBG=IDP    4934    A
DBG=IDP   STP5     RJ     =XRSR=      RESTORE ORIGINAL REGISTERS                 DBG=IDP    4935    A
DBG=IDP                                                                           DBG=IDP    4936    A
DBG=IDP  *         HERE TO XEQ NORMAL INSTRUCTION.                                DBG=IDP    4937    A
DBG=IDP                                                                           DBG=IDP    4938    A
DBG=IDP   STP=IN   BSSZ   1            THIS WORD IS PLUGGED WITH THE INSTRUCTION DBG=IDP    4939    A
DBG=IDP  *                               TO BE XEQ, LEFT JUSTIFIED WITH NO-OP FILL DBG=IDP    4940    A
DBG=IDP                                                                           DBG=IDP    4941    A
DBG=IDP            RJ     SVR=        SAVE ORIGINAL REGISTERS                    DBG=IDP    4942    A
DBG=IDP            EQ     STP=X                                                  DBG=IDP    4943    A
DBG=IDP                                                                           DBG=IDP    4944    A
DBG=IDP  *         HERE IF A BRANCH INSTRUCTION (01 THRU 07).                     DBG=IDP    4945    A
DBG=IDP                                                                           DBG=IDP    4946    A
DBG=IDP   STP6     SA2    STP=I                                                  DBG=IDP    4947    A
DBG=IDP            SX6    X1-01B                                                 DBG=IDP    4948    A
DBG=IDP            SA3    STP=ADR                                                DBG=IDP    4949    A
DBG=IDP            NZ     X6,STP8     IF NOT 01B INSTRUCTION                     DBG=IDP    4950    A
DBG=IDP            NZ     X2,STP5     IF NOT *RJ*                                DBG=IDP    4951    A
DBG=IDP                                                                           DBG=IDP    4952    A
DBG=IDP  *         HERE IF AN *RJ* INSTRUCTION.                                   DBG=IDP    4953    A
DBG=IDP                                                                           DBG=IDP    4954    A
DBG=IDP            SA4    IDPPREG                                                DBG=IDP    4955    A
DBG=IDP            SA5    ST=ENDX     (X5) = WORD CONTAINING *RJ* INSTRUCTION    DBG=IDP    4956    A
DBG=IDP  *                                (IF PREVIOUS CONTENTS OF GEN BRK)      DBG=IDP    4957    A
DBG=IDP            BX6    X4          (X6) = /*+1/                               DBG=IDP    4958    A
DBG=IDP            SX7    X3-REG=                                                DBG=IDP    4959    A
DBG=IDP            SX0    X3-SNP=                                                DBG=IDP    4960    A
DBG=IDP            NZ     X5,STP6A    IF *RJ* IS PREVIOUS CONTENTS OF BREAK ADDR DBG=IDP    4961    A
DBG=IDP            SA5    X4          (X5) = WORD CONTAINING *RJ* INSTRUCTION    DBG=IDP    4962    A
DBG=IDP            SX6    X4+B1       (X6) = /*+1/                               DBG=IDP    4963    A
DBG=IDP                                                                           DBG=IDP    4964    A
DBG=IDP  *         CHECK FOR CALL TO *REG=/SNP=*.  THESE ROUTINES MUST AVOID      DBG=IDP    4965    A
DBG=IDP  *           BEING STEPPED BECAUSE THEY USE *SVR=/RSR=* TO SAVE AND       DBG=IDP    4966    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *          RESTORE THE CALLERS REGISTERS (SEE NEXT PARAGRAPH).
DBG=IDP
DBG=IDP   STP6A    ZR     X7,STP7     IF *RJ REG=*
DBG=IDP            ZR     X0,STP7     IF *RJ SNP=*
DBG=IDP
DBG=IDP  *         CHECK FOR CALL TO *RSR=/SVR=*.  THIS IS CURRENTLY FATALLY
DBG=IDP  *          RECURSIVE BECAUSE *STP* USES *RSR=/SVR=* TO SAVE AND RESTORE
DBG=IDP  *          THE ORIGINAL REGISTERS BEFORE AND AFTER EACH INSTRUCTION IS
DBG=IDP  *          STEPPED.  THIS IMPLIES THAT *RSR=/SVR=* WOULD HAVE TO BE
DBG=IDP  *          ABLE TO BE STEPPED AND EXECUTED CONCURRENTLY...TRICKY.
DBG=IDP
DBG=IDP            SX7    =XRSR=
DBG=IDP            SX0    =XSVR=
DBG=IDP            IX7    X3-X7
DBG=IDP            IX0    X3-X0
DBG=IDP            ZR     X7,STP=ER   IF *RJ RSR=*
DBG=IDP            ZR     X0,STP=ER   IF *RJ SVR=*
DBG=IDP
DBG=IDP  *         SET UP TRANSFER ADDRESS OF /RJ Q/ TO BE /Q+1/.
DBG=IDP
DBG=IDP            SX7    X3+B1       (X7) = /Q+1/
DBG=IDP            SB2    X7
DBG=IDP            SA7    IDPXFR
DBG=IDP
DBG=IDP  *         CHECK TRANSFER ADDR AND SIMULATE *RJ* BY PLUGGING /EQ *+1/.
DBG=IDP  *
DBG=IDP  *         ENTRY  (B2) = TRANSFER ADDRESS
DBG=IDP  *                (X6) = /*+1/  (I.E. RETURN ADDR)
DBG=IDP  *                (X3) = /Q/    (I.E. ADDR TO PLUG /EQ *+1/)
DBG=IDP  *
DBG=IDP  *         NOTE   WHEN THIS IS NOT *RJ REG=/SNP=*, THEN
DBG=IDP  *                TRANSFER ADDR = Q+1  OR  (B2) = (X3)+1.
DBG=IDP  *
DBG=IDP  *                WHEN THIS IS AN *RJ REG=/SNP=*, THEN
DBG=IDP  *                TRANSFER ADDR = *+1  OR  (B2) = (X6), I.E. THE *RJ* HAS
DBG=IDP  *                BEEN SKIPPED.
DBG=IDP
DBG=IDP   STP6B    RJ     CHK         CHECK CM ADDRESS
DBG=IDP            SX7    0400B       (X7) = *EQ* INSTRUCTION
DBG=IDP            MI     B2,STP=ER   IF ADDR IS BAD
DBG=IDP            LX7    29-11
DBG=IDP            SX0    IDP=
DBG=IDP            BX7    X7+X6       (X7) = 30/0,30/EQ *+1
DBG=IDP            LX7    59-29       (X7) = 30/EQ *+1,30/0
DBG=IDP            SA7    X3
DBG=IDP
DBG=IDP  *         CHECK FOR *RJ IDP=*.
DBG=IDP
DBG=IDP            IX0    X3-X0
DBG=IDP            SA1    IDPXRJ
DBG=IDP            ZR     X0,STP7A    IF *RJ IDP=*
DBG=IDP
```

```
         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *         SET UP LEVEL 0 *RJ* ADDRESS.                                  DBG=IDP    5019    A
DBG=IDP                                                                          DBG=IDP    5020    A
DBG=IDP            NZ     X1,STP=X    IF NOT AT LEVEL 0 *RJ*                     DBG=IDP    5021    A
DBG=IDP            SX6    X3                                                     DBG=IDP    5022    A
DBG=IDP            SA6    A1                                                     DBG=IDP    5023    A
DBG=IDP            EQ     STP=X                                                  DBG=IDP    5024    A
DBG=IDP                                                                          DBG=IDP    5025    A
DBG=IDP  *         HERE TO PROCESS *RJ REG=/SNP=*.                               DBG=IDP    5026    A
DBG=IDP                                                                          DBG=IDP    5027    A
DBG=IDP   STP7     SA6    IDPXFR      SET TO *RJ REG=/SNP= BECOMES               DBG=IDP    5028    A
DBG=IDP  *                              BRANCH TO /*+1/*                         DBG=IDP    5029    A
DBG=IDP            BX7    X5          (X7) = 30/RJ REG=/SNP=, 30/FWA OF PARM LIST DBG=IDP    5030    A
DBG=IDP            SA7    STP=SNP                                                DBG=IDP    5031    A
DBG=IDP            RJ     =XRSR=      RESTORE ORIGINAL REGISTERS                 DBG=IDP    5032    A
DBG=IDP                                                                          DBG=IDP    5033    A
DBG=IDP   STP=SNP  BSSZ   1           THIS WORD IS PLUGGED WITH AN *RJ REG=* OR  DBG=IDP    5034    A
DBG=IDP  *                              AN *RJ SNP=* FORCED UPPER WITH THE FWA OF DBG=IDP    5035    A
DBG=IDP  *                              THE PARAMETER LIST IN THE LOWER 18 BITS  DBG=IDP    5036    A
DBG=IDP                                                                          DBG=IDP    5037    A
DBG=IDP            SA1    IDPXFR      (X1) = ADDR TO XFER TO (I.E. /*+1/)        DBG=IDP    5038    A
DBG=IDP            SA3    STP=ADR     (X3) = *REG=* OR *SNP=*                    DBG=IDP    5039    A
DBG=IDP            SB2    X1                                                     DBG=IDP    5040    A
DBG=IDP            BX6    X1                                                     DBG=IDP    5041    A
DBG=IDP            EQ     STP6B                                                  DBG=IDP    5042    A
DBG=IDP                                                                          DBG=IDP    5043    A
DBG=IDP  *         HERE TO PROCESS *RJ IDP=*.                                    DBG=IDP    5044    A
DBG=IDP                                                                          DBG=IDP    5045    A
DBG=IDP   STP7A    SA6    IDPXFR      SET TO */RJ IDP=/ BECOMES BRANCH TO /*+1/* DBG=IDP    5046    A
DBG=IDP            SX1    A5+B1       (X1) = ADDR+1 OF *RJ IDP=*                 DBG=IDP    5047    A
DBG=IDP            SX5    X5          (X5) = FWA OF PARAMETER LIST (IF .NZ.)     DBG=IDP    5048    A
DBG=IDP            RJ     BRK         BREAK PROCESSOR                            DBG=IDP    5049    A
DBG=IDP            ZR     X5,STP=X    IF NO BREAK THIS TIME                      DBG=IDP    5050    A
DBG=IDP            RJ     FRK         CHECK FREQUENCY PARAMETERS                 DBG=IDP    5051    A
DBG=IDP            ZR     X5,STP=X    IF NO BREAK THIS TIME                      DBG=IDP    5052    A
DBG=IDP                                                                          DBG=IDP    5053    A
DBG=IDP  *         HERE TO HONOR BREAK.                                          DBG=IDP    5054    A
DBG=IDP                                                                          DBG=IDP    5055    A
DBG=IDP            SA1    IDPXFR                                                 DBG=IDP    5056    A
DBG=IDP            SX7    60                                                     DBG=IDP    5057    A
DBG=IDP            BX6    X1                                                     DBG=IDP    5058    A
DBG=IDP            SA7    IDPPOS                                                 DBG=IDP    5059    A
DBG=IDP            SA6    IDPPREG                                                DBG=IDP    5060    A
DBG=IDP            EQ     IDP2A       HONOR THIS BREAK...                        DBG=IDP    5061    A
DBG=IDP                                                                          DBG=IDP    5062    A
DBG=IDP  *         HERE IF BRANCH INSTRUCTION (02 THRU 07).                      DBG=IDP    5063    A
DBG=IDP                                                                          DBG=IDP    5064    A
DBG=IDP   STP8     SX6    X1-4                                                   DBG=IDP    5065    A
DBG=IDP            SX7    X1-2                                                   DBG=IDP    5066    A
DBG=IDP            ZR     X6,STP10    IF *EQ BI,BJ,ADDR*                         DBG=IDP    5067    A
DBG=IDP            ZR     X7,STP11    IF *JP BI+ADDR*                            DBG=IDP    5068    A
DBG=IDP                                                                          DBG=IDP    5069    A
DBG=IDP  *         HERE IF CONDITIONAL BRANCH INSTRUCTION.                       DBG=IDP    5070    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP                                                                                  DBG=IDP     5071     A
DBG=IDP   STP9     SA1    STP=IN                                                         DBG=IDP     5072     A
DBG=IDP            MX0    -18                                                            DBG=IDP     5073     A
DBG=IDP            LX0    48-18                                                          DBG=IDP     5074     A
DBG=IDP            SX2    STP=BP      (X2) = BRANCH *PASS* ADDR                          DBG=IDP     5075     A
DBG=IDP            LX2    48-18                                                          DBG=IDP     5076     A
DBG=IDP            BX1    X0*X1                                                          DBG=IDP     5077     A
DBG=IDP            BX2    -X0*X2                                                         DBG=IDP     5078     A
DBG=IDP            IX6    X1+X2       (X6) = 12/BRANCH OPC, 18/STP=BRP, 30/NO-OPS        DBG=IDP     5079     A
DBG=IDP            SA6    STP=BR                                                         DBG=IDP     5080     A
DBG=IDP                                                                                  DBG=IDP     5081     A
DBG=IDP   *        HERE TO XEQ CONDITIONAL BRANCH INSTRUCTION. THE 18 BIT ADDRESS        DBG=IDP     5082     A
DBG=IDP   *          *Q* PORTION OF THE BRANCH INSTRUCTION IS MODIFIED SO THAT IF        DBG=IDP     5083     A
DBG=IDP   *          THE BRANCH IS TAKEN, CONTROL PASSES TO *STP=BP*.                    DBG=IDP     5084     A
DBG=IDP                                                                                  DBG=IDP     5085     A
DBG=IDP            RJ     =XRSR=      RESTORE ORIGINAL REGISTERS                         DBG=IDP     5086     A
DBG=IDP                                                                                  DBG=IDP     5087     A
DBG=IDP   STP=BR   BSSZ   1           BRANCH TO XEQ IS PLUGGED HERE -- LEFT JUST-        DBG=IDP     5088     A
DBG=IDP   *                             IFIED W/ *Q* ALTERED AND NO-OP FILL              DBG=IDP     5089     A
DBG=IDP                                                                                  DBG=IDP     5090     A
DBG=IDP   STP=BF   SB1    1           RESTORE (B1) = 1                                   DBG=IDP     5091     A
DBG=IDP            EQ     STP=X                                                          DBG=IDP     5092     A
DBG=IDP                                                                                  DBG=IDP     5093     A
DBG=IDP   STP=BP   SA1    STP=ADR                                                        DBG=IDP     5094     A
DBG=IDP            SB1    1           RESTORE (B1) = 1                                   DBG=IDP     5095     A
DBG=IDP            SB2    X1          (B2) = ADDR TO TRANSFER CONTROL TO                 DBG=IDP     5096     A
DBG=IDP            SX6    X1                                                             DBG=IDP     5097     A
DBG=IDP            SA6    IDPXFR                                                         DBG=IDP     5098     A
DBG=IDP            RJ     CHK         CHECK CM ADDRESS                                   DBG=IDP     5099     A
DBG=IDP            SA1    IDPXLST                                                        DBG=IDP     5100     A
DBG=IDP            MI     B2,STP=ER   IF ADDR IS BAD                                     DBG=IDP     5101     A
DBG=IDP            ZR     X1,STP=X    IF NOT LISTING                                     DBG=IDP     5102     A
DBG=IDP            PRIDP  (=C=                              BRANCH TAKEN...=)             DBG=IDP     5103     A
DBG=IDP            EQ     STP=X                                                          DBG=IDP     5104     A
DBG=IDP                                                                                  DBG=IDP     5105     A
DBG=IDP   *        HERE IF *EQ BI,BJ,ADDR*.                                              DBG=IDP     5106     A
DBG=IDP                                                                                  DBG=IDP     5107     A
DBG=IDP   STP10    SA1    STP=I                                                          DBG=IDP     5108     A
DBG=IDP            SA2    STP=J                                                          DBG=IDP     5109     A
DBG=IDP            BX6    X1+X2                                                          DBG=IDP     5110     A
DBG=IDP            NZ     X6,STP9     IF NOT *EQ ADDR*                                   DBG=IDP     5111     A
DBG=IDP            MX7    0                                                              DBG=IDP     5112     A
DBG=IDP            SA1    IDPXRJ                                                         DBG=IDP     5113     A
DBG=IDP            SA2    IDPPREG                                                        DBG=IDP     5114     A
DBG=IDP            IX6    X1-X2                                                          DBG=IDP     5115     A
DBG=IDP            NZ     X6,STP=BP   IF NOT POSSIBLY *RJ SUBR* EXIT                     DBG=IDP     5116     A
DBG=IDP            SA7    A1          SET TO *BACK TO LEVEL 0 NOW*                       DBG=IDP     5117     A
DBG=IDP            EQ     STP=BP                                                         DBG=IDP     5118     A
DBG=IDP                                                                                  DBG=IDP     5119     A
DBG=IDP   *        HERE IF *JP BI+ADDR*.                                                 DBG=IDP     5120     A
DBG=IDP                                                                                  DBG=IDP     5121     A
DBG=IDP   STP11    SA1    STP=I                                                          DBG=IDP     5122     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            SA2    STP=ADR                                                  DBG=IDP    5123    A
DBG=IDP            SA3    =XSV=B+X1    (X3) = (BI)                                 DBG=IDP    5124    A
DBG=IDP            IX6    X2+X3        (X6) = BI+ADDR                              DBG=IDP    5125    A
DBG=IDP            SB2    X6                                                       DBG=IDP    5126    A
DBG=IDP            SA6    IDPXFR                                                   DBG=IDP    5127    A
DBG=IDP            RJ     CHK          CHECK CM ADDRESS                            DBG=IDP    5128    A
DBG=IDP            PL     B2,STP=X     IF ADDR OK                                  DBG=IDP    5129    A
DBG=IDP            EQ     STP=ER                                                   DBG=IDP    5130    A
DBG=IDP                                                                            DBG=IDP    5131    A
DBG=IDP  *         HERE IF A LOAD/STORE INSTRUCTION (50 THRU 57).                  DBG=IDP    5132    A
DBG=IDP                                                                            DBG=IDP    5133    A
DBG=IDP   STP12    SA1    STP=LS+X1-50B  (X1) = LOAD/STORE SKELETON                DBG=IDP    5134    A
DBG=IDP            SA2    STP=I                                                    DBG=IDP    5135    A
DBG=IDP            SA3    A2+B1                                                    DBG=IDP    5136    A
DBG=IDP            SA4    A3+B1                                                    DBG=IDP    5137    A
DBG=IDP            SA5    STP=ADR                                                  DBG=IDP    5138    A
DBG=IDP            SB6    X1           (B6) = .MI. IF *Q* FLAG, ELSE               DBG=IDP    5139    A
DBG=IDP  *                                   = FWA OF APPROPRIATE SAVED REG BLOCK  DBG=IDP    5140    A
DBG=IDP            LX1    -18-3                                                    DBG=IDP    5141    A
DBG=IDP            BX6    X1                                                       DBG=IDP    5142    A
DBG=IDP            AX6    59           (X6) = +0 IF *J* REG + *K* REG              DBG=IDP    5143    A
DBG=IDP  *                                       -0 IF *J* REG - *K* REG           DBG=IDP    5144    A
DBG=IDP            SB7    X1                                                       DBG=IDP    5145    A
DBG=IDP            ZR     X2,STP5      IF *SA0* (I.E. NOT LOAD/STORE)              DBG=IDP    5146    A
DBG=IDP            MI     B6,STP13     IF *Q* 18 BIT ADDR                          DBG=IDP    5147    A
DBG=IDP            SA5    B6+X4        (X5) = SAVED *K* REGISTER                   DBG=IDP    5148    A
DBG=IDP            BX5    X5-X6                                                    DBG=IDP    5149    A
DBG=IDP                                                                            DBG=IDP    5150    A
DBG=IDP   STP13    SA1    B7+X3        (X1) = SAVED *J* REGISTER                   DBG=IDP    5151    A
DBG=IDP            IX6    X1+X5                                                    DBG=IDP    5152    A
DBG=IDP            SB2    X6           (B2) = ADDR TO LOAD/STORE                   DBG=IDP    5153    A
DBG=IDP            SA6    IDPXAR                                                   DBG=IDP    5154    A
DBG=IDP            RJ     =XCHK        CHECK CM ADDRESS                            DBG=IDP    5155    A
DBG=IDP            PL     B2,STP5      IF ADDR OK                                  DBG=IDP    5156    A
DBG=IDP            EQ     STP=ER                                                   DBG=IDP    5157    A
DBG=IDP                                                                            DBG=IDP    5158    A
DBG=IDP  *         LOAD/STORE TABLE.                                               DBG=IDP    5159    A
DBG=IDP                                                                            DBG=IDP    5160    A
DBG=IDP   STP=LS   BSS    0                                                        DBG=IDP    5161    A
DBG=IDP                                                                            DBG=IDP    5162    A
DBG=IDP            LOC    50B                                                      DBG=IDP    5163    A
DBG=IDP   +        VFD    18/0,3/0,18/=XSV=A,3/0,18/-1         SAI  AJ+Q           DBG=IDP    5164    A
DBG=IDP   +        VFD    18/0,3/0,18/=XSV=B,3/0,18/-1         SAI  BJ+Q           DBG=IDP    5165    A
DBG=IDP   +        VFD    18/0,3/0,18/=XSV=X,3/0,18/-1         SAI  XJ+Q           DBG=IDP    5166    A
DBG=IDP   +        VFD    18/0,3/0,18/=XSV=X,3/0,18/=XSV=B     SAI  XJ+BK          DBG=IDP    5167    A
DBG=IDP   +        VFD    18/0,3/0,18/=XSV=A,3/0,18/=XSV=B     SAI  AJ+BK          DBG=IDP    5168    A
DBG=IDP   +        VFD    18/0,3/0,18/=XSV=A,3/4,18/=XSV=B     SAI  AJ-BK          DBG=IDP    5169    A
DBG=IDP   +        VFD    18/0,3/0,18/=XSV=B,3/0,18/=XSV=B     SAI  BJ+BK          DBG=IDP    5170    A
DBG=IDP   +        VFD    18/0,3/0,18/=XSV=B,3/4,18/=XSV=B     SAI  BJ-BK          DBG=IDP    5171    A
DBG=IDP            LOC    *O                                                       DBG=IDP    5172    A
DBG=IDP                                                                            DBG=IDP    5173    A
DBG=IDP  *         HERE TO UPDATE POS COUNTER AND PSEUDO P REGISTER.               DBG=IDP    5174    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP                                                                                  DBG=IDP     5175     A
DBG=IDP   STP=X    SA1    STP=OP                                                         DBG=IDP     5176     A
DBG=IDP            SA2    IDPXFR                                                         DBG=IDP     5177     A
DBG=IDP            SA3    IDPPREG                                                        DBG=IDP     5178     A
DBG=IDP            SA4    IDPPOS                                                         DBG=IDP     5179     A
DBG=IDP            SA5    ST=ENDX                                                        DBG=IDP     5180     A
DBG=IDP            SA1    DAZ=PS+X1    (X1) = INSTRUCTION SKELETON                       DBG=IDP     5181     A
DBG=IDP            SB5    15                                                             DBG=IDP     5182     A
DBG=IDP            BX6    X2                                                             DBG=IDP     5183     A
DBG=IDP            SB4    X4                                                             DBG=IDP     5184     A
DBG=IDP            MX7    0                                                              DBG=IDP     5185     A
DBG=IDP            PL     X1,STP=X2    IF SHORT (15 BIT) INSTRUCTION                     DBG=IDP     5186     A
DBG=IDP            SB5    B5+B5                                                          DBG=IDP     5187     A
DBG=IDP                                                                                  DBG=IDP     5188     A
DBG=IDP   STP=X2   LX1    59-57                                                          DBG=IDP     5189     A
DBG=IDP            NZ     X2,STP=X4A   IF A XFER OF CONTROL OCCURRED                     DBG=IDP     5190     A
DBG=IDP            SX6    X3+B1                                                          DBG=IDP     5191     A
DBG=IDP            LE     B4,B5,STP=X3  IF NO MORE INSTRUCTIONS IN THIS WORD             DBG=IDP     5192     A
DBG=IDP            PL     X1,STP=X5    IF INSTRUCTION DOES NOT FORCE UPPER               DBG=IDP     5193     A
DBG=IDP                                                                                  DBG=IDP     5194     A
DBG=IDP   STP=X3   ZR     X5,STP=X4    IF NOT IN IDP GENERATED BREAK MODE                DBG=IDP     5195     A
DBG=IDP            SX6    X3                                                             DBG=IDP     5196     A
DBG=IDP                                                                                  DBG=IDP     5197     A
DBG=IDP  *         HERE IF INSTRUCTION FORCES UPPER.                                     DBG=IDP     5198     A
DBG=IDP                                                                                  DBG=IDP     5199     A
DBG=IDP   STP=X4   SA7    A5           SET TO *NOT IN IDP GENERATED BREAK MODE*          DBG=IDP     5200     A
DBG=IDP            SA6    A3                                                             DBG=IDP     5201     A
DBG=IDP            SX7    60                                                             DBG=IDP     5202     A
DBG=IDP            SA7    A4                                                             DBG=IDP     5203     A
DBG=IDP            EQ     STP=X6                                                         DBG=IDP     5204     A
DBG=IDP                                                                                  DBG=IDP     5205     A
DBG=IDP  *         HERE IF TRANSFER OF CONTROL OCCURRED -- ENTER TRANSFER ADDR           DBG=IDP     5206     A
DBG=IDP  *           ONTO PUSH DOWN STACK OF SAVED TRANSFER ADDRESSES *IDPXFT*.          DBG=IDP     5207     A
DBG=IDP  *           ALL ENTRIES IN PUSH DOWN STACK ARE MOVED UP (TOWARD *RA*)           DBG=IDP     5208     A
DBG=IDP  *           ONE POSITION (THE 1ST ENTRY IN *XFT*, THE EARLIEST TRANSFER         DBG=IDP     5209     A
DBG=IDP  *           ADDRESS, FALLS OFF STACK).  EXAMPLE --                              DBG=IDP     5210     A
DBG=IDP  *                                                                               DBG=IDP     5211     A
DBG=IDP  *                     BEFORE                  AFTER                             DBG=IDP     5212     A
DBG=IDP  *             XFT+0  XFER(N)         XFT+0  XFER(N+1)                          DBG=IDP     5213     A
DBG=IDP  *                +1  XFER(N+1)          +1  XFER(N+2)                          DBG=IDP     5214     A
DBG=IDP  *                +2  XFER(N+2)          +2  XFER(N+3)                          DBG=IDP     5215     A
DBG=IDP  *                +3  XFER(N+3)          +3  XFER(N+4) -- NEW ENTRY             DBG=IDP     5216     A
DBG=IDP  *                +4  END-OF-TABLE       +4  END-OF-TABLE                       DBG=IDP     5217     A
DBG=IDP                                                                                  DBG=IDP     5218     A
DBG=IDP   STP=X4A  SA1    IDPXFT+1     (A1,X1) = A+C OF 1ST WORD TO MOVE                 DBG=IDP     5219     A
DBG=IDP            SB7    L.XFT-1      (B7) = NR OF WORDS TO MOVE                        DBG=IDP     5220     A
DBG=IDP                                                                                  DBG=IDP     5221     A
DBG=IDP   STP=X4B  BX7    X1                                                             DBG=IDP     5222     A
DBG=IDP            SA7    A1-B1                                                          DBG=IDP     5223     A
DBG=IDP            SB7    B7-B1                                                          DBG=IDP     5224     A
DBG=IDP            SA1    A1+B1                                                          DBG=IDP     5225     A
DBG=IDP            GT     B7,B0,STP=X4B IF NOT DONE                                      DBG=IDP     5226     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP                                                                                  DBG=IDP     5227     A
DBG=IDP            LX4    59-29                                                          DBG=IDP     5228     A
DBG=IDP            BX7    X4+X3       (X7) = 30/POS COUNTER,30/P REG OF XFER INST        DBG=IDP     5229     A
DBG=IDP            SA7    A1-B1       MAKE NEW ENTRY                                     DBG=IDP     5230     A
DBG=IDP            MX7    0                                                              DBG=IDP     5231     A
DBG=IDP            EQ     STP=X4      FORCE UPPER...                                     DBG=IDP     5232     A
DBG=IDP                                                                                  DBG=IDP     5233     A
DBG=IDP  *         HERE IF INSTRUCTION DOES NOT FORCE UPPER.                             DBG=IDP     5234     A
DBG=IDP                                                                                  DBG=IDP     5235     A
DBG=IDP   STP=X5   SB6    B4-B5                                                          DBG=IDP     5236     A
DBG=IDP            SX7    B6                                                             DBG=IDP     5237     A
DBG=IDP            SA7    A4                                                             DBG=IDP     5238     A
DBG=IDP                                                                                  DBG=IDP     5239     A
DBG=IDP  *         LIST RESULT REGISTER.                                                 DBG=IDP     5240     A
DBG=IDP                                                                                  DBG=IDP     5241     A
DBG=IDP   STP=X6   SA3    IDPXLST                                                        DBG=IDP     5242     A
DBG=IDP            SA4    STP=OP                                                         DBG=IDP     5243     A
DBG=IDP            ZR     X3,EXIT.    IF NOT LISTING                                     DBG=IDP     5244     A
DBG=IDP                                                                                  DBG=IDP     5245     A
DBG=IDP            SX6    X4-10B                                                         DBG=IDP     5246     A
DBG=IDP            SA3    STP=I                                                          DBG=IDP     5247     A
DBG=IDP            MI     X6,EXIT.    IF A BRANCH INSTRUCTION                            DBG=IDP     5248     A
DBG=IDP            LX1    -2          RESTORE (X1) = INSTRUCTION SKELETON                DBG=IDP     5249     A
DBG=IDP            MX0    -CHAR                                                          DBG=IDP     5250     A
DBG=IDP            BX6    -X0*X1                                                         DBG=IDP     5251     A
DBG=IDP            SX7    X6-1RB                                                         DBG=IDP     5252     A
DBG=IDP            SB6    000B        (B6) = 0TR, WHERE T=0 (B REG)                      DBG=IDP     5253     A
DBG=IDP            ZR     X7,STP=X8   IF B REGISTER                                      DBG=IDP     5254     A
DBG=IDP            SX7    X6-1RA                                                         DBG=IDP     5255     A
DBG=IDP            SB6    010B        (B6) = 0TR, WHERE T=1 (A REG)                      DBG=IDP     5256     A
DBG=IDP            ZR     X7,STP=X8   IF A REGISTER                                      DBG=IDP     5257     A
DBG=IDP            SX7    X6-1RX                                                         DBG=IDP     5258     A
DBG=IDP            SB6    020B        (B6) = 0TR, WHERE T=2 (X REG)                      DBG=IDP     5259     A
DBG=IDP            ZR     X7,STP=X9   IF X REGISTER                                      DBG=IDP     5260     A
DBG=IDP            EQ     EXIT.                                                          DBG=IDP     5261     A
DBG=IDP                                                                                  DBG=IDP     5262     A
DBG=IDP   STP=X8   SB4    B6+X3       (B4) = 0TR                                         DBG=IDP     5263     A
DBG=IDP            RJ     DAB         DUMP -A- OR -B- REGISTER                           DBG=IDP     5264     A
DBG=IDP            EQ     EXIT.                                                          DBG=IDP     5265     A
DBG=IDP                                                                                  DBG=IDP     5266     A
DBG=IDP   STP=X9   SB4    B6+X3       (B4) = 02R                                         DBG=IDP     5267     A
DBG=IDP            RJ     DUX         DUMP -X- REGISTER                                  DBG=IDP     5268     A
DBG=IDP            EQ     EXIT.                                                          DBG=IDP     5269     A
DBG=IDP                                                                                  DBG=IDP     5270     A
DBG=IDP  *         HERE IF AN ERROR ENCOUNTERED.                                         DBG=IDP     5271     A
DBG=IDP                                                                                  DBG=IDP     5272     A
DBG=IDP   STP=ER   SA1    IDPXLST                                                        DBG=IDP     5273     A
DBG=IDP            SA2    ST=ENDX                                                        DBG=IDP     5274     A
DBG=IDP            SA3    IDPPREG                                                        DBG=IDP     5275     A
DBG=IDP            SA4    IDPPOS                                                         DBG=IDP     5276     A
DBG=IDP            NZ     X1,STP=E3   IF INSTRUCTION WAS ALREADY LISTED                  DBG=IDP     5277     A
DBG=IDP            SA5    X3                                                             DBG=IDP     5278     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            ZR     X2,STP=E2   IF NOT IN IDP GENERATED BREAK MODE          DBG=IDP   5279   A
DBG=IDP            SA5    A2                                                       DBG=IDP   5280   A
DBG=IDP                                                                            DBG=IDP   5281   A
DBG=IDP   STP=E2   SB4    X4                                                       DBG=IDP   5282   A
DBG=IDP            RJ     DAZ         DEASSEMBLE BAD INSTRUCTION                   DBG=IDP   5283   A
DBG=IDP                                                                            DBG=IDP   5284   A
DBG=IDP   STP=E3   MX0    -18                                                      DBG=IDP   5285   A
DBG=IDP            SA1    IDPPREG                                                  DBG=IDP   5286   A
DBG=IDP            SA2    =10H      P =                                            DBG=IDP   5287   A
DBG=IDP            BX1    -X0*X1                                                   DBG=IDP   5288   A
DBG=IDP            LX7    X2                                                       DBG=IDP   5289   A
DBG=IDP            SA7    SNAPLNE                                                  DBG=IDP   5290   A
DBG=IDP            RJ     =XCOD       CONVERT BINARY TO OCTAL DPC                  DBG=IDP   5291   A
DBG=IDP            BX6    X4                                                       DBG=IDP   5292   A
DBG=IDP            SA6    A7+B1                                                    DBG=IDP   5293   A
DBG=IDP                                                                            DBG=IDP   5294   A
DBG=IDP   #FRA     IF     DEF,FRA=                                                 DBG=IDP   5295   A
DBG=IDP            SA1    IDPPREG                                                  DBG=IDP   5296   A
DBG=IDP            SB7    STP=E4      (B7) = RETURN ADDR FOR *FRA=*                DBG=IDP   5297   A
DBG=IDP            EQ     =XFRA=      FIND RELATIVE ADDRESS                        DBG=IDP   5298   A
DBG=IDP                                                                            DBG=IDP   5299   A
DBG=IDP   STP=E4   SA6    A6+B1                                                    DBG=IDP   5300   A
DBG=IDP            SA7    A6+B1                                                    DBG=IDP   5301   A
DBG=IDP                                                                            DBG=IDP   5302   A
DBG=IDP   #FRA     ELSE                                                             DBG=IDP   5303   A
DBG=IDP            MX6    0                                                        DBG=IDP   5304   A
DBG=IDP            SA6    A6+B1       MARK EOL                                     DBG=IDP   5305   A
DBG=IDP   #FRA     ENDIF                                                            DBG=IDP   5306   A
DBG=IDP                                                                            DBG=IDP   5307   A
DBG=IDP            PRIDP  SNAPLNE                                                  DBG=IDP   5308   A
DBG=IDP            EQ     IDP=ER                                                   DBG=IDP   5309   A
DBG=IDP                                                                            DBG=IDP   5310   A
DBG=IDP                                                                            DBG=IDP   5311   A
DBG=IDP   STP=ADR  =      DAZ=ADR                                                  DBG=IDP   5312   A
DBG=IDP   STP=I    =      DAZ=I                                                    DBG=IDP   5313   A
DBG=IDP   STP=J    =      DAZ=J                                                    DBG=IDP   5314   A
DBG=IDP   STP=K    =      DAZ=K                                                    DBG=IDP   5315   A
DBG=IDP   STP=OP   =      DAZ=OP                                                   DBG=IDP   5316   A
DBG=IDP   VFD      SPACE  4,8                                                      DBG=IDP   5317   A
DBG=IDP  **        VFD - VARIABLE FIELD DEFINITION.                                DBG=IDP   5318   A
DBG=IDP  *                                                                         DBG=IDP   5319   A
DBG=IDP  *                                                                         DBG=IDP   5320   A
DBG=IDP  *                CONVERTS A SPECIFIED FIELD IN A BINARY WORD TO OCTAL     DBG=IDP   5321   A
DBG=IDP  *         DISPLAY CODE WITH APPROPRIATE SPACING AND BLANK FILL.           DBG=IDP   5322   A
DBG=IDP  *         THE FIELD TO BE CONVERTED CAN BE REPRESENTED IN *COMPASS* BY--  DBG=IDP   5323   A
DBG=IDP  *                                                                         DBG=IDP   5324   A
DBG=IDP  *         POS    (B4)   IN (X1)                                           DBG=IDP   5325   A
DBG=IDP  *         VFD    (B5)/FIELD   WHERE *FIELD* HAS BEEN EXTRACTED FROM (X1)  DBG=IDP   5326   A
DBG=IDP  *                                                                         DBG=IDP   5327   A
DBG=IDP  *         E.G.   (B4) = 45   POSITION COUNTER                             DBG=IDP   5328   A
DBG=IDP  *                (B5) = 30   NR OF BITS                                   DBG=IDP   5329   A
DBG=IDP  *                                                                         DBG=IDP   5330   A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *                BINARY      76211031100015620310                          DBG=IDP     5331     A
DBG=IDP  *                DPC RSULT   .....0311000156.....    (.=BLANK(55B))        DBG=IDP     5332     A
DBG=IDP  *                                                                          DBG=IDP     5333     A
DBG=IDP  *         ENTRY  (X1)  =  BINARY WORD TO BE CONVERTED                      DBG=IDP     5334     A
DBG=IDP  *                (B4)  =  POSITION COUNTER                                 DBG=IDP     5335     A
DBG=IDP  *                (B5)  =  NR OF BITS IN FIELD                              DBG=IDP     5336     A
DBG=IDP  *                                                                          DBG=IDP     5337     A
DBG=IDP  *         EXIT   (X6)  =  CONVERTED UPPER 30 BITS OF (X1) -- DPC RESULT    DBG=IDP     5338     A
DBG=IDP  *                (X7)  =  CONVERTED LOWER 30 BITS OF (X1) -- DPC RESULT    DBG=IDP     5339     A
DBG=IDP  *                (X0)  =  .ZR. IF BAD POS OR BIT COUNT ON ENTRY,           DBG=IDP     5340     A
DBG=IDP  *                            ELSE .NZ.                                     DBG=IDP     5341     A
DBG=IDP  *                                                                          DBG=IDP     5342     A
DBG=IDP  *         USES   X - 0,1,2,3,4,6,7                                         DBG=IDP     5343     A
DBG=IDP  *                A - NONE                                                  DBG=IDP     5344     A
DBG=IDP  *                B - 6                                                     DBG=IDP     5345     A
DBG=IDP  *                                                                          DBG=IDP     5346     A
DBG=IDP  *         CALLS  NONE                                                      DBG=IDP     5347     A
DBG=IDP                                                                             DBG=IDP     5348     A
DBG=IDP                                                                             DBG=IDP     5349     A
DBG=IDP   VFD      SUBR              ** ENTRY/EXIT **                               DBG=IDP     5350     A
DBG=IDP            MX3    0                                                         DBG=IDP     5351     A
DBG=IDP            BX0    X0-X0                                                     DBG=IDP     5352     A
DBG=IDP            SB6    B5-1                                                      DBG=IDP     5353     A
DBG=IDP            LT     B4,B0,EXIT. IF POS COUNT IS BAD                           DBG=IDP     5354     A
DBG=IDP            LT     B5,B0,EXIT. IF BIT COUNT IS BAD                           DBG=IDP     5355     A
DBG=IDP            MX0    -3                                                        DBG=IDP     5356     A
DBG=IDP            ZR     B5,VFD2     IF NOT CONVERTING ANY BITS                    DBG=IDP     5357     A
DBG=IDP            MX3    1                                                         DBG=IDP     5358     A
DBG=IDP            AX3    X3,B6                                                     DBG=IDP     5359     A
DBG=IDP            NO                                                               DBG=IDP     5360     A
DBG=IDP            LX3    X3,B4       (X3) = EXTRACT MASK FOR BITS TO CONVERT       DBG=IDP     5361     A
DBG=IDP            BX1    X3*X1                                                     DBG=IDP     5362     A
DBG=IDP                                                                             DBG=IDP     5363     A
DBG=IDP   VFD2     MX6    0                                                         DBG=IDP     5364     A
DBG=IDP            BX7    X7-X7                                                     DBG=IDP     5365     A
DBG=IDP            SB6    60                                                        DBG=IDP     5366     A
DBG=IDP                                                                             DBG=IDP     5367     A
DBG=IDP  *         ASSEMBLE APPROPRIATE DIGITS.                                     DBG=IDP     5368     A
DBG=IDP                                                                             DBG=IDP     5369     A
DBG=IDP   VFD3     LX1    3                                                         DBG=IDP     5370     A
DBG=IDP            BX2    -X0*X1                                                    DBG=IDP     5371     A
DBG=IDP            LX3    3                                                         DBG=IDP     5372     A
DBG=IDP            BX4    -X0*X3                                                    DBG=IDP     5373     A
DBG=IDP            SX2    X2+1R0                                                    DBG=IDP     5374     A
DBG=IDP            SB6    B6-6                                                      DBG=IDP     5375     A
DBG=IDP            NZ     X4,VFD4     IF ASSEMBLING THESE DIGITS                    DBG=IDP     5376     A
DBG=IDP            SX2    1R                                                        DBG=IDP     5377     A
DBG=IDP                                                                             DBG=IDP     5378     A
DBG=IDP   VFD4     LX2    X2,B6                                                     DBG=IDP     5379     A
DBG=IDP            BX7    X7+X2                                                     DBG=IDP     5380     A
DBG=IDP            GT     B6,B0,VFD3  IF PACKING REG (X7) NOT FULL                  DBG=IDP     5381     A
DBG=IDP                                                                             DBG=IDP     5382     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *         HERE IF PACKING REGISTER (X7) IS FULL.                    DBG=IDP   5383   A
DBG=IDP                                                                      DBG=IDP   5384   A
DBG=IDP            SB6    60                                                 DBG=IDP   5385   A
DBG=IDP            NZ     X6,EXIT.    IF FINISHED ASSEMBLING ENTIRE WORD      DBG=IDP   5386   A
DBG=IDP            BX6    X7                                                 DBG=IDP   5387   A
DBG=IDP            MX7    0                                                  DBG=IDP   5388   A
DBG=IDP            EQ     VFD3                                               DBG=IDP   5389   A
DBG=IDP   DATA     SPACE  4,8                                                DBG=IDP   5390   A
DBG=IDP  **        *IDP* DATA DECLARATIONS.                                  DBG=IDP   5391   A
DBG=IDP                                                                      DBG=IDP   5392   A
DBG=IDP                                                                      DBG=IDP   5393   A
DBG=IDP   IDPSTMT  BSSZ   1           DURING TOKEN GENERATION *BUB* -- CONTAINS  DBG=IDP   5394   A
DBG=IDP  *                              ADDR OF STMT INFO WORD FOR STMT BEING  DBG=IDP   5395   A
DBG=IDP  *                              BURST/BUILT, SO THAT UPON ENCOUNTERING DBG=IDP   5396   A
DBG=IDP  *                              EOS, *BUB* CAN GO BACK AND FILL IN LEN. DBG=IDP   5397   A
DBG=IDP  *                            DURING STMT PROCESSING -- INDICATES WHERE DBG=IDP   5398   A
DBG=IDP  *                              STMT BEING PROCESSED IS.               DBG=IDP   5399   A
DBG=IDP  *                              IF STMT IS NOT PART OF *PROC* -- BIT 59 DBG=IDP   5400   A
DBG=IDP  *                                IS 0 AND BITS 17 THRU 0 CONTAIN THE  DBG=IDP   5401   A
DBG=IDP  *                                ADDR OF STMT INFO WORD IN *IDPTB* FOR DBG=IDP   5402   A
DBG=IDP  *                                THIS STMT.                           DBG=IDP   5403   A
DBG=IDP  *                              IF STMT IS PART OF *PROC* -- BIT 59 IS 1 DBG=IDP   5404   A
DBG=IDP  *                                AND BITS 17 THRU 0 CONTAIN THE *IDPPRN* DBG=IDP   5405   A
DBG=IDP  *                                ORD OF *PROC* CONTAINING STMT.       DBG=IDP   5406   A
DBG=IDP                                                                      DBG=IDP   5407   A
DBG=IDP   IDPSTO   BSSZ   3                                                  DBG=IDP   5408   A
DBG=IDP                                                                      DBG=IDP   5409   A
DBG=IDP            LOC    IDPSTO                                             DBG=IDP   5410   A
DBG=IDP   IDPSTA   =      *           ADDR TO STORE INTO FOR *STORE* PROCESSING DBG=IDP   5411   A
DBG=IDP   IDPSTC   =      *+1         NEW VALUE ASSEMBLED FROM C1 THRU C4 ON DBG=IDP   5412   A
DBG=IDP  *                              *STORE* COMMAND                      DBG=IDP   5413   A
DBG=IDP   IDPSTM   =      *+2         MASK ASSEMBLED FROM NULL C1 THRU C4 ON DBG=IDP   5414   A
DBG=IDP  *                              *STORE* COMMAND. IDP USES THIS MASK TO DBG=IDP   5415   A
DBG=IDP  *                              PICK UP THE PREVIOUS CONTENTS OF THE C-N- DBG=IDP   5416   A
DBG=IDP  *                              FIELD IF THE NEW C-N- FIELD IS NULL  DBG=IDP   5417   A
DBG=IDP  *                              E.G.  STORE ADDR,,22222,,44444       DBG=IDP   5418   A
DBG=IDP  *                              PREVIOUS (STA) = 01020304050607080910 DBG=IDP   5419   A
DBG=IDP  *                                       (STC) = 00000222220000044444 DBG=IDP   5420   A
DBG=IDP  *                                       (STM) = 77777000007777700000 DBG=IDP   5421   A
DBG=IDP  *                                   NEW (STA) = 01020222220607044444 DBG=IDP   5422   A
DBG=IDP            LOC    *O                                                 DBG=IDP   5423   A
DBG=IDP                                                                      DBG=IDP   5424   A
DBG=IDP   IDPBAX   BSS    0           BAX REGISTER SHIFT MASK                DBG=IDP   5425   A
DBG=IDP            ECHO   2,BIT=(A,B,X)                                      DBG=IDP   5426   A
DBG=IDP            POS    60-1R_BIT                                          DBG=IDP   5427   A
DBG=IDP            VFD    1/1                                                DBG=IDP   5428   A
DBG=IDP            POS    0                                                  DBG=IDP   5429   A
DBG=IDP                                                                      DBG=IDP   5430   A
DBG=IDP   IDPSMX   BSS    0           SEPARATOR TOKEN MASK +-*,              DBG=IDP   5431   A
DBG=IDP            POS    60-1R+                                             DBG=IDP   5432   A
DBG=IDP            VFD    1/1                                                DBG=IDP   5433   A
DBG=IDP            POS    60-1R-                                             DBG=IDP   5434   A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            VFD    1/1                                                  DBG=IDP     5435     A
DBG=IDP            POS    60-1R*                                               DBG=IDP     5436     A
DBG=IDP            VFD    1/1                                                  DBG=IDP     5437     A
DBG=IDP            POS    60-1R,                                               DBG=IDP     5438     A
DBG=IDP            VFD    1/1                                                  DBG=IDP     5439     A
DBG=IDP            POS    0                                                    DBG=IDP     5440     A
DBG=IDP   IDPBTC   BSSZ   1           BREAK TYPE CODE                          DBG=IDP     5441     A
DBG=IDP                                                                        DBG=IDP     5442     A
DBG=IDP   IDPACC   BSSZ   1           ACCUMULATOR FOR THIS SUBEXPRESSION       DBG=IDP     5443     A
DBG=IDP   IDPIND   BSSZ   1           USED BY *PAS* TO SHOW THE LEVEL OF INDIR-  DBG=IDP     5444     A
DBG=IDP   *                             ECT ADDRESSING, 0 = DIRECT ADDRESSING  DBG=IDP     5445     A
DBG=IDP   IDPNUL   BSSZ   1           .ZR. IF JUST PARSED NULL EXPRESSION      DBG=IDP     5446     A
DBG=IDP   *                                 E.G.   SNAP FWA,,LEN   (LWA IS NULL)  DBG=IDP     5447     A
DBG=IDP   IDPSGN   BSSZ   1           SET TO +0 WHEN + SIGN ENCOUNTERED        DBG=IDP     5448     A
DBG=IDP   *                                 -0 WHEN - SIGN ENCOUNTERED         DBG=IDP     5449     A
DBG=IDP                                                                        DBG=IDP     5450     A
DBG=IDP   IDP0TR   BSSZ   1           REGISTER DESIGNATOR                      DBG=IDP     5451     A
DBG=IDP   *                             USED FOR *REG R1,R2,...,RN* PROCESSING.  DBG=IDP     5452     A
DBG=IDP   *                             FORMAT IS--  60/0TR WHERE              DBG=IDP     5453     A
DBG=IDP   *                                 T = REGISTER TYPE (B=0,A=1,X=2)    DBG=IDP     5454     A
DBG=IDP   *                                 R = REGISTER NR (0-7)              DBG=IDP     5455     A
DBG=IDP                                                                        DBG=IDP     5456     A
DBG=IDP   IDPNAM   BSSZ   1           SAVE CELL FOR *ST=SET* -- SET NAME (0L FMT)  DBG=IDP     5457     A
DBG=IDP                                                                        DBG=IDP     5458     A
DBG=IDP   IDPFAD   BSSZ   1           .NZ. IF *DAZ* IS TO FORCE OUT ADDR OF INST-  DBG=IDP     5459     A
DBG=IDP   *                             RUCTION BEING DEASSEMBLED EVEN IF NOT  DBG=IDP     5460     A
DBG=IDP   *                             FORCED UPPER IN WORD, ELSE .ZR.        DBG=IDP     5461     A
DBG=IDP   *                             E.G. IF *FAD* IS .ZR. --               DBG=IDP     5462     A
DBG=IDP   *                             ADDR  54111                SA1  A1+B1  DBG=IDP     5463     A
DBG=IDP   *                                       0311000000       NZ   X1,...  DBG=IDP     5464     A
DBG=IDP   *                                                 10611  BX6  X1     DBG=IDP     5465     A
DBG=IDP   *                             E.G. IF *FAD* IS .NZ. --               DBG=IDP     5466     A
DBG=IDP   *                             ADDR  54111                SA1  A1+B1  DBG=IDP     5467     A
DBG=IDP   *                             ADDR       0311000000      NZ   X1,...  DBG=IDP     5468     A
DBG=IDP   *                             ADDR                10611  BX6  X1     DBG=IDP     5469     A
DBG=IDP   *                             NOTE-- *DAZ* WILL ALWAYS ZERO *FAD*    DBG=IDP     5470     A
DBG=IDP                                                                        DBG=IDP     5471     A
DBG=IDP   IDPSA5   BSSZ   1           SAVE CELL FOR (A5)                       DBG=IDP     5472     A
DBG=IDP                                                                        DBG=IDP     5473     A
DBG=IDP   IDPPPP   BSSZ   2           PSEUDO P REGISTER AND POS COUNTER        DBG=IDP     5474     A
DBG=IDP                                                                        DBG=IDP     5475     A
DBG=IDP            LOC    IDPPPP                                               DBG=IDP     5476     A
DBG=IDP   IDPPREG  =      *           PSEUDO P REGISTER. POINTS TO WORD CONTAIN-  DBG=IDP     5477     A
DBG=IDP   *                             ING NEXT INSTRUCTION TO BE XEQ         DBG=IDP     5478     A
DBG=IDP   IDPPOS   =      *+1         POSITION COUNTER. LEFT MOST BIT OF INST-  DBG=IDP     5479     A
DBG=IDP   *                             RUCTION TO XEQ IS BIT (IDPPOS)-1 IN    DBG=IDP     5480     A
DBG=IDP   *                             ((IDPPREG)).(BITS ARE NUMBERED 59 THRU 0)  DBG=IDP     5481     A
DBG=IDP            LOC    *O                                                   DBG=IDP     5482     A
DBG=IDP                                                                        DBG=IDP     5483     A
DBG=IDP   IDPXAR   BSSZ   1           .MI. IF NO ADDR REF VIA LOAD/STORE, ELSE DBG=IDP     5484     A
DBG=IDP   *                             ADDR THAT WAS REFERENCED. USED IN *STAR*  DBG=IDP     5485     A
DBG=IDP   *                             PROCESSING.                            DBG=IDP     5486     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP   IDPXCA   BSSZ   2          CONDITIONAL *STEP* COMPARE ADDRESSES.          DBG=IDP   5487   A
DBG=IDP   *                            IF THIS IS A CONDITIONAL STEP (STNE,...),    DBG=IDP   5488   A
DBG=IDP   *                            THEN ((XA1)) WILL BE COMPARED TO ((XA2))     DBG=IDP   5489   A
DBG=IDP                                                                             DBG=IDP   5490   A
DBG=IDP            LOC    IDPXCA                                                    DBG=IDP   5491   A
DBG=IDP   IDPXA1   =      *                                                         DBG=IDP   5492   A
DBG=IDP   IDPXA2   =      *+1                                                       DBG=IDP   5493   A
DBG=IDP            LOC    *O                                                        DBG=IDP   5494   A
DBG=IDP                                                                             DBG=IDP   5495   A
DBG=IDP   IDPXFR   BSSZ   1          .ZR. IF NO XFER OF CONTROL BY BRANCH OR RJ,    DBG=IDP   5496   A
DBG=IDP   *                            ELSE ADDR TO XFER CONTROL TO. USED IN        DBG=IDP   5497   A
DBG=IDP   *                            STEP MODE SO THAT PSEUDO P REGISTER IS       DBG=IDP   5498   A
DBG=IDP   *                            NOT PREMATURELY UPDATED WHEN BRANCHING.      DBG=IDP   5499   A
DBG=IDP   IDPXLST  BSSZ   1          .NZ. IF LISTING THIS INSTRUCTION IN STEP       DBG=IDP   5500   A
DBG=IDP   *                            MODE, ELSE .ZR.                              DBG=IDP   5501   A
DBG=IDP   IDPXRJ   BSSZ   1          .ZR. IF AT LEVEL 0 (I.E. NO *RJ* ACTIVE),      DBG=IDP   5502   A
DBG=IDP   *                            ELSE ADDR OF ENTRY POINT TO ROUTINE          DBG=IDP   5503   A
DBG=IDP   *                            CALLED FROM LEVEL 0. USED IN STEP MODE TO    DBG=IDP   5504   A
DBG=IDP   *                            DETERMINE WHEN TO TURN LISTING BACK ON       DBG=IDP   5505   A
DBG=IDP   *                            WHEN /-RJ/ OPTION SELECTED                   DBG=IDP   5506   A
DBG=IDP   IDPXTC   BSSZ   1          *STEP* TYPE CODE                               DBG=IDP   5507   A
DBG=IDP                                                                             DBG=IDP   5508   A
DBG=IDP   IDPXOP   BSSZ   2          STEP OPTIONS FLAGS. THERE IS A ONE-TO-ONE      DBG=IDP   5509   A
DBG=IDP   *                            RELATIONSHIP BETWEEN THE STEP OPTIONS        DBG=IDP   5510   A
DBG=IDP   *                            FLAGS AND THE STEP OPTIONS KEYWORD TABLE.    DBG=IDP   5511   A
DBG=IDP   *                            WHEN *POL* RETURNS WITH A KEYWORD FIND       DBG=IDP   5512   A
DBG=IDP   *                            AND AN OPTIONS FLAG VALUE, THE ORDINAL OF    DBG=IDP   5513   A
DBG=IDP   *                            FIND IN THE STEP OPTIONS KEYWORD TABLE IS    DBG=IDP   5514   A
DBG=IDP   *                            THE ORDINAL TO STORE INTO *IDPXOP*.          DBG=IDP   5515   A
DBG=IDP                                                                             DBG=IDP   5516   A
DBG=IDP            LOC    IDPXOP                                                    DBG=IDP   5517   A
DBG=IDP   IDPXOL   =      *          .NZ. IF /L/ SELECTED ON STEP COMAND,           DBG=IDP   5518   A
DBG=IDP   *                            ELSE .ZR. IF /-L/ SELECTED                   DBG=IDP   5519   A
DBG=IDP   IDPXORJ  =      *+1        .NZ. IF /RJ/ SELECTED ON STEP COMMAND,         DBG=IDP   5520   A
DBG=IDP   *                            ELSE .ZR. IF /-RJ/ SELECTED                  DBG=IDP   5521   A
DBG=IDP            LOC    *O                                                        DBG=IDP   5522   A
DBG=IDP   MX=      SPACE  4,8                                                       DBG=IDP   5523   A
DBG=IDP   **       MX= - CHARACTER SHIFT MASKS.                                     DBG=IDP   5524   A
DBG=IDP                                                                             DBG=IDP   5525   A
DBG=IDP                                                                             DBG=IDP   5526   A
DBG=IDP   MX=KEYW  CHARMX (A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P,Q,R,S,T,U,V,W,X,Y,Z,0,    DBG=IDP   5527   A
DBG=IDP   ,1,2,3,4,5,6,7,8,9),+7777B-1RM                                            DBG=IDP   5528   A
DBG=IDP   MX=NR09  CHARMX (0,1,2,3,4,5,6,7,8,9)                                     DBG=IDP   5529   A
DBG=IDP   MX=TOKN  CHARMX ("EOS",+,-,*,(,)),+7777B-1RM                              DBG=IDP   5530   A
DBG=IDP   MSG      SPACE  4,8                                                       DBG=IDP   5531   A
DBG=IDP   **       *IDP* MESSAGES.                                                  DBG=IDP   5532   A
DBG=IDP                                                                             DBG=IDP   5533   A
DBG=IDP   IDPMSG2  DIS    3, CALLD BY 000000 IN XXXXXXX                             DBG=IDP   5534   A
DBG=IDP   IDPMSG4  DIS    3, OLD VALU 00000000000000000000                          DBG=IDP   5535   A
DBG=IDP            DATA   0                                                         DBG=IDP   5536   A
DBG=IDP   IDPMSG5  DIS    3, NEW VALU 00000000000000000000                          DBG=IDP   5537   A
DBG=IDP            DATA   0                                                         DBG=IDP   5538   A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP   PATFLG   SPACE  4,8                                          001980   DBG=IDP     5539     A
DBG=IDP  **        PATFLG - PARSE *FWA,LWA,LEN* TRIPLE FLAG.           001990   DBG=IDP     5540     A
DBG=IDP  *                                                             002000   DBG=IDP     5541     A
DBG=IDP  *                                                             002010   DBG=IDP     5542     A
DBG=IDP  *         CONTAINS STATUS ABOUT THE *FWA,LWA,LEN* TRIPLE BEING 002020   DBG=IDP     5543     A
DBG=IDP  *         PARSED/PROCESSED BY ROUTINE *PAT*.  USED SO THAT *PAT* 002030   DBG=IDP     5544     A
DBG=IDP  *         CAN RESOLVE ANY POSSIBLE SYNTACTIC DIFFERENCES/PROBLEMS 002040   DBG=IDP     5545     A
DBG=IDP  *         THAT MIGHT OCCUR BETWEEN THE *LWA* AND *LEN* PARAMETERS. 002050   DBG=IDP     5546     A
DBG=IDP  *         SEE SUBROUTINE *PAT*.                               002060   DBG=IDP     5547     A
DBG=IDP  *                                                             002070   DBG=IDP     5548     A
DBG=IDP  *         BIT FLAGS WITHIN *PATFLG* MUST EXIST IN BITS 0 THRU 16 002080   DBG=IDP     5549     A
DBG=IDP  *         (I.E. FIELD MANIPULATION IS DONE VIA *SX.I* TYPE INSTRUCTIONS)002090   DBG=IDP     5550     A
DBG=IDP  *         AND ARE DESCRIBE/DEFINED IN THE USUAL MANNER VIA SYMBOLS 002100   DBG=IDP     5551     A
DBG=IDP  *         OF THE FORM --                                      002110   DBG=IDP     5552     A
DBG=IDP  *                                                             002120   DBG=IDP     5553     A
DBG=IDP  *               PAF.XXXP  AND  PAF.XXXL                       002130   DBG=IDP     5554     A
DBG=IDP  *                                                             002140   DBG=IDP     5555     A
DBG=IDP  *         WHERE *PAF.* IS A COMMON PREFIX AND *XXX* IS A UNIQUE 002150   DBG=IDP     5556     A
DBG=IDP  *         BIT FLAG (FIELD) NAME.                              002160   DBG=IDP     5557     A
DBG=IDP  *                                                             002170   DBG=IDP     5558     A
DBG=IDP  *         BIT FLAGS *XXX* ARE --                              002180   DBG=IDP     5559     A
DBG=IDP  *                                                             002190   DBG=IDP     5560     A
DBG=IDP  *         LWA  = 1 IF LWA WAS SPECIFIED EXPLICITLY (I.E. NON-NULL), 002200   DBG=IDP     5561     A
DBG=IDP  *                ELSE 0.                                      002210   DBG=IDP     5562     A
DBG=IDP  *         LEN  = 1 IF LEN WAS SPECIFIED EXPLICITLY (I.E. NON-NULL), 002220   DBG=IDP     5563     A
DBG=IDP  *                ELSE 0.                                      002230   DBG=IDP     5564     A
DBG=IDP                                                                002240   DBG=IDP     5565     A
DBG=IDP                                                                002250   DBG=IDP     5566     A
DBG=IDP   PATFLG   BSSZ   1                                            002260   DBG=IDP     5567     A
DBG=IDP   F.IDI    SPACE  4,8                                                    DBG=IDP     5568     A
DBG=IDP  **        F.IDI - FET, LINE IMAGE AREA, AND BUFFER FOR INTERACTIVE      DBG=IDP     5569     A
DBG=IDP  *           DEBUG INPUT FILE.                                           DBG=IDP     5570     A
DBG=IDP                                                                          DBG=IDP     5571     A
DBG=IDP   #IDI     IF     -DEF,F.IDI                                             DBG=IDP     5572     A
DBG=IDP   L.IDI    =      101B        LENGTH OF *IDP* INPUT BUFFER              DBG=IDP     5573     A
DBG=IDP                                                                          DBG=IDP     5574     A
DBG=IDP   F.IDI    BSS    0           ** FWA OF FET **                          DBG=IDP     5575     A
DBG=IDP   ZZZZZDI  FILEC  IDPIDI,L.IDI,(FET=7)                           000190   DBG=IDP     5576     A
DBG=IDP            DIS    1, **IDP**                                            DBG=IDP     5577     A
DBG=IDP   IDPLIM   DIS    8,INTERACTIVE DEBUG LINE IMAGE AREA.                  DBG=IDP     5578     A
DBG=IDP   IDPIDI   BSS    L.IDI                                                 DBG=IDP     5579     A
DBG=IDP                                                                          DBG=IDP     5580     A
DBG=IDP   L=LIM    BSSZ   1           NR OF WORDS IN SOURCE LINE IMAGE AT       DBG=IDP     5581     A
DBG=IDP  *                              (IDPLIM) ET SEQ (INCLUDES WORD CONTAINING DBG=IDP     5582     A
DBG=IDP  *                              EOL MARK)                                DBG=IDP     5583     A
DBG=IDP   #IDI     ENDIF                                                         DBG=IDP     5584     A
DBG=IDP   F.BDI    SPACE  4,8                                                    DBG=IDP     5585     A
DBG=IDP   F.BDI    =      F.IDI                                                  DBG=IDP     5586     A
DBG=IDP   F.IDO    SPACE  4,8                                                    DBG=IDP     5587     A
DBG=IDP  **        F.IDO - FET AND BUFFER FOR INTERACTIVE DEBUG OUTPUT FILE.     DBG=IDP     5588     A
DBG=IDP                                                                          DBG=IDP     5589     A
DBG=IDP   #IDO     IF     -DEF,F.IDO                                             DBG=IDP     5590     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP   L.IDO    =      101B        LENGTH OF *IDP* OUTPUT BUFFER                    DBG=IDP    5591    A
DBG=IDP                                                                                DBG=IDP    5592    A
DBG=IDP   F.IDO    BSS    0           ** FWA OF FET **                                 DBG=IDP    5593    A
DBG=IDP   ZZZZZDO  FILEC  IDPIDO,L.IDO,(FET=7)                            000210       DBG=IDP    5594    A
DBG=IDP   IDPIDO   BSS    L.IDO                                                        DBG=IDP    5595    A
DBG=IDP   #IDO     ENDIF                                                               DBG=IDP    5596    A
DBG=IDP   IDPTB    SPACE  4,8                                                         DBG=IDP    5597    A
DBG=IDP  **        IDPTB - COMMAND LINE TOKEN BUFFER.                                 DBG=IDP    5598    A
DBG=IDP  *                                                                            DBG=IDP    5599    A
DBG=IDP  *                CONTAINS THE CURRENT COMMAND LINE IN TOKEN FORM.            DBG=IDP    5600    A
DBG=IDP  *                                                                            DBG=IDP    5601    A
DBG=IDP  *         TOKEN BUFFER FORMAT --                                             DBG=IDP    5602    A
DBG=IDP  *                                                                            DBG=IDP    5603    A
DBG=IDP  *         VFD    1/LAS,41/0,18/LEN   (TOKEN INFO WORD FOR STMT 1)            DBG=IDP    5604    A
DBG=IDP  *         VFD    42/0LCHARS,18/TOKEN TYPE                                    DBG=IDP    5605    A
DBG=IDP  *         VFD    42/0LCHAR ,18/TOKEN TYPE                                    DBG=IDP    5606    A
DBG=IDP  *          .                                                                 DBG=IDP    5607    A
DBG=IDP  *          .                                                                 DBG=IDP    5608    A
DBG=IDP  *         VFD    60/0   (END OF STATEMENT MARK FOR STMT 1)                   DBG=IDP    5609    A
DBG=IDP  *                                                                            DBG=IDP    5610    A
DBG=IDP  *         VFD    1/LAS,41/0,18/LEN   (TOKEN INFO WORD FOR STMT 2)            DBG=IDP    5611    A
DBG=IDP  *          .                                                                 DBG=IDP    5612    A
DBG=IDP  *         ETC                                                                DBG=IDP    5613    A
DBG=IDP  *                                                                            DBG=IDP    5614    A
DBG=IDP  *                THE FIELDS IN THE TOKEN INFORMATION WORDS ARE DESCRIBED     DBG=IDP    5615    A
DBG=IDP  *         BY PAIRS OF SYMBOLS OF THE FORM *TB.XXXP* AND *TB.XXXL*,           DBG=IDP    5616    A
DBG=IDP  *         WHERE *TB* IS THE COMMON PREFIX, AND *XXX* IS THE FIELD            DBG=IDP    5617    A
DBG=IDP  *         DESCRIPTION.                                                       DBG=IDP    5618    A
DBG=IDP  *                                                                            DBG=IDP    5619    A
DBG=IDP  *         FIELD DESCRIPTIONS (XXX) ARE --                                    DBG=IDP    5620    A
DBG=IDP  *                                                                            DBG=IDP    5621    A
DBG=IDP  *         LAS  = 1  IF THIS IS LAST STMT ON LINE, ELSE 0                     DBG=IDP    5622    A
DBG=IDP  *                                                                            DBG=IDP    5623    A
DBG=IDP  *         LEN  = NR OF WORDS OR TOKENS IN STMT  (INCLUDES EOL MARK)          DBG=IDP    5624    A
DBG=IDP  *                  NOTE--IF LAS .EQ. 0, TOKEN INFO WORD FOR NEXT             DBG=IDP    5625    A
DBG=IDP  *                  STMT IS AT -- TOKEN INFO WORD + LEN + 1                   DBG=IDP    5626    A
DBG=IDP                                                                               DBG=IDP    5627    A
DBG=IDP                                                                               DBG=IDP    5628    A
DBG=IDP   L.TB     =      40          LENGTH OF TOKEN BUFFER                          DBG=IDP    5629    A
DBG=IDP   IDPTB    BSSZ   L.TB                                                        DBG=IDP    5630    A
DBG=IDP   TOKEN    SPACE  4,8                                                         DBG=IDP    5631    A
DBG=IDP  **        IDP TOKEN VALUES.                                                  DBG=IDP    5632    A
DBG=IDP                                                                               DBG=IDP    5633    A
DBG=IDP   O.CONS   =      2           CONSTANT                                        DBG=IDP    5634    A
DBG=IDP   O.SEP    =      4           SEPARATOR +-*,                                  DBG=IDP    5635    A
DBG=IDP   O.VAR    =      3           NAME                                            DBG=IDP    5636    A
DBG=IDP   KEYW     SPACE  4,8                                                         DBG=IDP    5637    A
DBG=IDP  **        KEYW - *IDP* KEYWORD TABLE GENERATOR.                              DBG=IDP    5638    A
DBG=IDP  *                                                                            DBG=IDP    5639    A
DBG=IDP  *                                                                            DBG=IDP    5640    A
DBG=IDP  *                MACRO USED TO GENERATE THE *IDP* COMMAND KEYWORD TABLE.     DBG=IDP    5641    A
DBG=IDP  *                                                                            DBG=IDP    5642    A
```

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *         KEYW   KEYWRD,ADDR                                                  DBG=IDP     5643     A
DBG=IDP  *                                                                             DBG=IDP     5644     A
DBG=IDP  *         ENTRY  *KEYWRD* =  THE *IDP* KEYWORD                                DBG=IDP     5645     A
DBG=IDP  *                *ADDR*   =  STATEMENT PROCESSOR FOR THIS KEYWORD             DBG=IDP     5646     A
DBG=IDP  *                                                                             DBG=IDP     5647     A
DBG=IDP  *         EXIT   NONE                                                         DBG=IDP     5648     A
DBG=IDP  *                                                                             DBG=IDP     5649     A
DBG=IDP  *         USES   NONE                                                         DBG=IDP     5650     A
DBG=IDP  *                                                                             DBG=IDP     5651     A
DBG=IDP  *         CALLS  NONE                                                         DBG=IDP     5652     A
DBG=IDP                                                                                DBG=IDP     5653     A
DBG=IDP                                                                                DBG=IDP     5654     A
DBG=IDP            PURGMAC  KEYW                                                       DBG=IDP     5655     A
DBG=IDP                                                                                DBG=IDP     5656     A
DBG=IDP   KEYW     MACRO  KEYWRD,ADDR                                                  DBG=IDP     5657     A
DBG=IDP            VFD    42/0L_KEYWRD,18/ADDR                                         DBG=IDP     5658     A
DBG=IDP   KEYW     ENDM                                                                DBG=IDP     5659     A
DBG=IDP   KEY      SPACE  4,8                                                          DBG=IDP     5660     A
DBG=IDP  **        *IDP* KEYWORD TABLE.                                                DBG=IDP     5661     A
DBG=IDP                                                                                DBG=IDP     5662     A
DBG=IDP   IDPKEY   BSS    0                                                            DBG=IDP     5663     A
DBG=IDP            KEYW   CODE,ST=COD                                                  DBG=IDP     5664     A
DBG=IDP            KEYW   CONNECT,ST=CON                                               DBG=IDP     5665     A
DBG=IDP            KEYW   DISCONT,ST=DSC                                               DBG=IDP     5666     A
DBG=IDP            KEYW   DPC,ST=DPC                                                   DBG=IDP     5667     A
DBG=IDP            KEYW   END,ST=END                                                   DBG=IDP     5668     A
DBG=IDP            KEYW   FREEZE,ST=FRZ                                     002360     DBG=IDP     5669     A
DBG=IDP            KEYW   JUMP,ST=JP                                                   DBG=IDP     5670     A
DBG=IDP            KEYW   OUTPUT,ST=OUT                                                DBG=IDP     5671     A
DBG=IDP            KEYW   REGS,ST=REG                                                  DBG=IDP     5672     A
DBG=IDP            KEYW   RESET,ST=RES                                                 DBG=IDP     5673     A
DBG=IDP            KEYW   SET,ST=SET                                                   DBG=IDP     5674     A
DBG=IDP            KEYW   SNAP,ST=SNP                                                  DBG=IDP     5675     A
DBG=IDP            KEYW   STORE,ST=STO                                                 DBG=IDP     5676     A
DBG=IDP            KEYW   UNBREAK,ST=UBK                                               DBG=IDP     5677     A
DBG=IDP            KEYW   UNSET,ST=UST                                                 DBG=IDP     5678     A
DBG=IDP            KEYW   UBREAK,ST=UBK                                                DBG=IDP     5679     A
DBG=IDP            KEYW   USET,ST=UST                                                  DBG=IDP     5680     A
DBG=IDP            KEYW   WHERE,ST=WHR                                                 DBG=IDP     5681     A
DBG=IDP            KEYW   XEQ,ST=XEQ                                                   DBG=IDP     5682     A
DBG=IDP            KEYW   XFER,ST=XFR                                                  DBG=IDP     5683     A
DBG=IDP                                                                                DBG=IDP     5684     A
DBG=IDP  *         *BREAK* KEYWORDS.                                                   DBG=IDP     5685     A
DBG=IDP  *             THE BREAK TYPE CODE (IDPBTC) IS DETERMINED FROM THE BREAK       DBG=IDP     5686     A
DBG=IDP  *           KEYWORD ORDINAL.                                                  DBG=IDP     5687     A
DBG=IDP  *             THERE EXISTS A ONE-TO-ONE RELATIONSHIP BETWEEN THE              DBG=IDP     5688     A
DBG=IDP  *           BREAK KEYWORD TABLE AND THE *CBC* OPERATOR TABLE. A CHANGE        DBG=IDP     5689     A
DBG=IDP  *           TO ONE WOULD AFFECT THE OTHER.                                    DBG=IDP     5690     A
DBG=IDP  *             THE ORDER OF THE BREAK KEYWORDS WITHIN *IDPKBR* IS ALSO         DBG=IDP     5691     A
DBG=IDP  *           SIGNIFICANT--                                                     DBG=IDP     5692     A
DBG=IDP  *               BTC = 0  IS A BREAK WITH ONLY ONE ADDRESS EXPRESSION--        DBG=IDP     5693     A
DBG=IDP  *                          BREAK ADDR                                         DBG=IDP     5694     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

DBG=IDP  *                BTC .LT. BTC=EQ  ARE BREAKS WITH 2 ADDR EXPRESSIONS--      DBG=IDP     5695     A
DBG=IDP  *                           BRPL ADDR,ADDR1                                 DBG=IDP     5696     A
DBG=IDP  *                BTC .GE. BTC=EQ  ARE BREAKS WITH 3 ADDR EXPRESSIONS--      DBG=IDP     5697     A
DBG=IDP  *                           BREQ ADDR,ADDR1,ADDR2                           DBG=IDP     5698     A
DBG=IDP  *           THIS INFORMATION IS USED THROUGHOUT BREAK PROCESSING.           DBG=IDP     5699     A
DBG=IDP                                                                              DBG=IDP     5700     A
DBG=IDP   IDPKBR   BSS    0                                                          DBG=IDP     5701     A
DBG=IDP            LOC    0                                                          DBG=IDP     5702     A
DBG=IDP            KEYW   BREAK,ST=BRK                                               DBG=IDP     5703     A
DBG=IDP            KEYW   BRPL,ST=BRK                                                DBG=IDP     5704     A
DBG=IDP            KEYW   BRMI,ST=BRK                                                DBG=IDP     5705     A
DBG=IDP            KEYW   BRZR,ST=BRK                                                DBG=IDP     5706     A
DBG=IDP            KEYW   BRNZ,ST=BRK                                                DBG=IDP     5707     A
DBG=IDP   BTC=EQ   KEYW   BREQ,ST=BRK                                                DBG=IDP     5708     A
DBG=IDP            KEYW   BRNE,ST=BRK                                                DBG=IDP     5709     A
DBG=IDP            KEYW   BRLT,ST=BRK                                                DBG=IDP     5710     A
DBG=IDP            KEYW   BRGE,ST=BRK                                                DBG=IDP     5711     A
DBG=IDP            KEYW   BRLE,ST=BRK                                                DBG=IDP     5712     A
DBG=IDP            KEYW   BRGT,ST=BRK                                                DBG=IDP     5713     A
DBG=IDP            LOC    *O                                                         DBG=IDP     5714     A
DBG=IDP                                                                              DBG=IDP     5715     A
DBG=IDP  *         *STEP* KEYWORDS.                                                  DBG=IDP     5716     A
DBG=IDP  *             THE STEP TYPE CODE (IDPXTC) IS DETERMINED FROM THE STEP       DBG=IDP     5717     A
DBG=IDP  *           KEYWORD ORDINAL.                                                DBG=IDP     5718     A
DBG=IDP  *             THERE EXISTS A ONE-TO-ONE RELATIONSHIP BETWEEN THE STEP       DBG=IDP     5719     A
DBG=IDP  *           KEYWORD TABLE AND THE *CBC* OPERATOR TABLE.                     DBG=IDP     5720     A
DBG=IDP  *             THE ORDER OF THE STEP KEYWORDS WITHIN *IDPKST* IS ALSO        DBG=IDP     5721     A
DBG=IDP  *           SIGNIFICANT--                                                   DBG=IDP     5722     A
DBG=IDP  *                XTC = 0  IS A STEP WITH NO ADDR EXPRESSION--               DBG=IDP     5723     A
DBG=IDP  *                           STEP L,RJ                                       DBG=IDP     5724     A
DBG=IDP  *                XTC .LT. XTC=EQ ARE STEPS WITH 1 ADDR EXPRESSION--         DBG=IDP     5725     A
DBG=IDP  *                           STMI ADDR,L,RJ                                  DBG=IDP     5726     A
DBG=IDP  *                XTC .GE. XTC=EQ AND                                        DBG=IDP     5727     A
DBG=IDP  *                    .LT. XTC=RNG ARE STEPS WITH 2 ADDR EXPRESSIONS--       DBG=IDP     5728     A
DBG=IDP  *                           STGT ADDR1,ADDR2,L,RJ                           DBG=IDP     5729     A
DBG=IDP  *                XTC .GE. XTC=RNG ARE STEPS THAT ARE SPECIAL CASED--        DBG=IDP     5730     A
DBG=IDP  *                           STAR ADDR,L,RJ                                  DBG=IDP     5731     A
DBG=IDP  *           (ALSO SEE *BREAK* KEYWORDS)                                     DBG=IDP     5732     A
DBG=IDP                                                                              DBG=IDP     5733     A
DBG=IDP   IDPKST   BSS    0                                                          DBG=IDP     5734     A
DBG=IDP            LOC    0                                                          DBG=IDP     5735     A
DBG=IDP            KEYW   STEP,ST=STP                                                DBG=IDP     5736     A
DBG=IDP            KEYW   STPL,ST=STP                                                DBG=IDP     5737     A
DBG=IDP            KEYW   STMI,ST=STP                                                DBG=IDP     5738     A
DBG=IDP            KEYW   STZR,ST=STP                                                DBG=IDP     5739     A
DBG=IDP            KEYW   STNZ,ST=STP                                                DBG=IDP     5740     A
DBG=IDP   XTC=EQ   KEYW   STEQ,ST=STP                                                DBG=IDP     5741     A
DBG=IDP            KEYW   STNE,ST=STP                                                DBG=IDP     5742     A
DBG=IDP            KEYW   STLT,ST=STP                                                DBG=IDP     5743     A
DBG=IDP            KEYW   STGE,ST=STP                                                DBG=IDP     5744     A
DBG=IDP            KEYW   STLE,ST=STP                                                DBG=IDP     5745     A
DBG=IDP            KEYW   STGT,ST=STP                                                DBG=IDP     5746     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP   XTC=RNG  KEYW   STRANGE,ST=STP                                  DBG=IDP    5747    A
DBG=IDP   XTC=NR   KEYW   STNR,ST=XNR                                     DBG=IDP    5748    A
DBG=IDP   XTC=AR   KEYW   STAR,ST=STP                                     DBG=IDP    5749    A
DBG=IDP            LOC    *O                                              DBG=IDP    5750    A
DBG=IDP            DATA   0          END OF TABLE MARK                    DBG=IDP    5751    A
DBG=IDP                                                                   DBG=IDP    5752    A
DBG=IDP  **        *STEP* OPTIONS KEYWORD TABLE.                          DBG=IDP    5753    A
DBG=IDP  *           THERE IS A ONE-TO-ONE RELATIONSHIP BETWEEN THE STEP OPTIONS  DBG=IDP  5754  A
DBG=IDP  *           KEYWORD TABLE AND THE STEP OPTIONS FLAGS (SEE *IDPXOP*)  DBG=IDP  5755  A
DBG=IDP                                                                   DBG=IDP    5756    A
DBG=IDP   IDPKSTO  BSS    0                                               DBG=IDP    5757    A
DBG=IDP            KEYW   L,1                                             DBG=IDP    5758    A
DBG=IDP            KEYW   RJ,1                                            DBG=IDP    5759    A
DBG=IDP            DATA   0          END OF TABLE MARK                    DBG=IDP    5760    A
DBG=IDP   KEY      SPACE  4,8                                             DBG=IDP    5761    A
DBG=IDP  **        *OUTPUT* OPTIONS KEYWORD TABLE.                        DBG=IDP    5762    A
DBG=IDP                                                                   DBG=IDP    5763    A
DBG=IDP   IDPKOUT  BSS    0                                               DBG=IDP    5764    A
DBG=IDP            KEYW   B,ST=OUT3                                       DBG=IDP    5765    A
DBG=IDP            KEYW   BECHO,ST=OUT4                                   DBG=IDP    5766    A
DBG=IDP            KEYW   I,ST=OUT2                                       DBG=IDP    5767    A
DBG=IDP            KEYW   IECHO,ST=OUT5                                   DBG=IDP    5768    A
DBG=IDP            DATA   0          END OF TABLE MARK                    DBG=IDP    5769    A
DBG=IDP   IDPBA    SPACE  4,8                                             DBG=IDP    5770    A
DBG=IDP  **        IDPBA/IDPBC - BREAK ADDRESS AND BREAK CONTENTS PARALLEL TBLS.  DBG=IDP  5771  A
DBG=IDP                                                                   DBG=IDP    5772    A
DBG=IDP   L.BA     =      10         LENGTH OF *IDPBA*                    DBG=IDP    5773    A
DBG=IDP   IDPBA    BSS    0          FWA OF *BREAK* ADDR TABLE            DBG=IDP    5774    A
DBG=IDP   .BA      DUP    L.BA,1                                          DBG=IDP    5775    A
DBG=IDP            CON    -1                                              DBG=IDP    5776    A
DBG=IDP            DATA   0          END OF TABLE TERMINATOR              DBG=IDP    5777    A
DBG=IDP   IDPBC    BSSZ   2*L.BA     *BREAK* ADDR CONTENTS TABLE          DBG=IDP    5778    A
DBG=IDP   IDPTMP   SPACE  4,8                                             DBG=IDP    5779    A
DBG=IDP  **        IDP USER *TEMP* TABLE.                                 DBG=IDP    5780    A
DBG=IDP                                                                   DBG=IDP    5781    A
DBG=IDP   L.TMP    =      10D                                             DBG=IDP    5782    A
DBG=IDP   IDPTMP   BSSZ   L.TMP                                           DBG=IDP    5783    A
DBG=IDP   IDPSET   SPACE  4,8                                             DBG=IDP    5784    A
DBG=IDP  **        IDP *SET* TABLE DEFINITION.                            DBG=IDP    5785    A
DBG=IDP                                                                   DBG=IDP    5786    A
DBG=IDP   L.SET    =      30B        LENGTH OF IDP *SET* TABLE            DBG=IDP    5787    A
DBG=IDP   IDPSET   BSS    0                                               DBG=IDP    5788    A
DBG=IDP            DUP    L.SET,1                                         DBG=IDP    5789    A
DBG=IDP            CON    -1                                              DBG=IDP    5790    A
DBG=IDP            DATA   0                                               DBG=IDP    5791    A
DBG=IDP   IDPXFT   SPACE  4,8                                             DBG=IDP    5792    A
DBG=IDP  **        IDPXFT - MOST RECENT TRANSFER ADDRESSES TABLE.         DBG=IDP    5793    A
DBG=IDP  *                                                                DBG=IDP    5794    A
DBG=IDP  *           TABLE FORMAT --                                      DBG=IDP    5795    A
DBG=IDP  *                                                                DBG=IDP    5796    A
DBG=IDP  *             XFT+0    VFD  30/-1  AVAILABLE ENTRY(ONLY IF XFT NOT FULL)  DBG=IDP  5797  A
DBG=IDP  *                        .                                       DBG=IDP    5798    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *                    .                                                  DBG=IDP    5799    A
DBG=IDP  *               +N    VFD  30/POS COUNTER,30/P REG OF EARLIEST XFER      DBG=IDP    5800    A
DBG=IDP  *               +N+1  VFD  30/POS COUNTER,30/P REG OF LATER XFER         DBG=IDP    5801    A
DBG=IDP  *                    .                                                  DBG=IDP    5802    A
DBG=IDP  *              +L.XFT VFD  60/0   END OF TABLE MARK                      DBG=IDP    5803    A
DBG=IDP                                                                           DBG=IDP    5804    A
DBG=IDP   L.XFT    =      4                                                       DBG=IDP    5805    A
DBG=IDP   IDPXFT   BSS    0                                                       DBG=IDP    5806    A
DBG=IDP            DUP    L.XFT,1                                                 DBG=IDP    5807    A
DBG=IDP            CON    -1                                                      DBG=IDP    5808    A
DBG=IDP            DATA   0                                                       DBG=IDP    5809    A
DBG=IDP   IFR      SPACE  4,8                                          002380     DBG=IDP    5810    A
DBG=IDP  ***       IFR - IDP FREEZE RESTART.                           002390     DBG=IDP    5811    A
DBG=IDP  *                                                             002400     DBG=IDP    5812    A
DBG=IDP  *                                                             002410     DBG=IDP    5813    A
DBG=IDP                                                                002420     DBG=IDP    5814    A
DBG=IDP   FW.IFR   BSS    0           MARK FWA OF *FREEZE* RESTART AREA 002430     DBG=IDP    5815    A
DBG=IDP                                                                002440     DBG=IDP    5816    A
DBG=IDP            QUAL   IFR                                          002450     DBG=IDP    5817    A
DBG=IDP   IFR      SPACE  4,8                                          002460     DBG=IDP    5818    A
DBG=IDP  **        IFR - IDP FREEZE RECOVERY/RESTART.                  002470     DBG=IDP    5819    A
DBG=IDP  *                                                             002480     DBG=IDP    5820    A
DBG=IDP  *                                                             002490     DBG=IDP    5821    A
DBG=IDP  *         THIS ROUTINE IS USED TO RESTART (I.E. SWAPIN) AN    002500     DBG=IDP    5822    A
DBG=IDP  *         IDP HOST PROGRAM THAT WAS FROZEN VIA *FREEZE*       002510     DBG=IDP    5823    A
DBG=IDP  *         COMMAND.  THE PROGRAM TO RESTART WAS WRITTEN BY *FRZ* 002520   DBG=IDP    5824    A
DBG=IDP  *         TO LFN *F.FRZ* IN A SPECIAL FORMAT --               002530     DBG=IDP    5825    A
DBG=IDP  *                                                             002540     DBG=IDP    5826    A
DBG=IDP  *           RECORD 1  CONTAINS *IFR* (THE CODE YOU ARE LOOKING AT) 002550 DBG=IDP    5827    A
DBG=IDP  *                     IN THE FORMAT OF AN ABSOLUTE BINARY.        002560     DBG=IDP    5828    A
DBG=IDP  *                                                             002570     DBG=IDP    5829    A
DBG=IDP  *           RECORD 2  CONTAINS THE CORE IMAGE OF THE FROZEN PROGRAM. 002580 DBG=IDP    5830    A
DBG=IDP  *                                                             002590     DBG=IDP    5831    A
DBG=IDP  *         BECAUSE RECORD 1 OF *F.FRZ* LOOKS LIKE AN ABS BINARY, THE 002600 DBG=IDP    5832    A
DBG=IDP  *         PROGRAMMER NEED ONLY ENTER THE NAME OF THE FREEZE FILE AT 002610 DBG=IDP    5833    A
DBG=IDP  *         HIS OR HER TERMINAL IN ORDER TO RESTART THE HOST.   002620     DBG=IDP    5834    A
DBG=IDP  *                                                             002630     DBG=IDP    5835    A
DBG=IDP  *         AFTER *IFR* HAS BEEN LOADED, IT WILL MAKE AN OPERATING SYSTEM 002640 DBG=IDP 5836 A
DBG=IDP  *         REQUEST FOR THE AMOUNT OF CM THAT THE HOST HAD AT THE TIME IT 002650 DBG=IDP 5837 A
DBG=IDP  *         WAS FROZEN PLUS ENOUGH FOR *IFR* TO PERFORM THE FOLLOWING 002660 DBG=IDP    5838    A
DBG=IDP  *         TASKS.  AFTER *IFR* HAS CREATED A HOLE FOR THE HOST 002670     DBG=IDP    5839    A
DBG=IDP  *         BY *MEM*ING, IT WILL *PLUG* SOME CODE UP ABOVE THE HOLE, 002680 DBG=IDP    5840    A
DBG=IDP  *         WHICH WILL READ IN THE 2ND RECORD OF *F.FRZ*.       002690     DBG=IDP    5841    A
DBG=IDP  *                                                             002700     DBG=IDP    5842    A
DBG=IDP  *         WE ALL HAVE CHET RICHARDS OF SVLOPS TO THANK FOR THIS 002710   DBG=IDP    5843    A
DBG=IDP  *         DIABOLICALLY CLEVER IDEA.  HI HO SILVER, AND AWAY...  002720   DBG=IDP    5844    A
DBG=IDP  *                                                             002730     DBG=IDP    5845    A
DBG=IDP  *         IMPORTANT NOTE--  IT IS IMPORTANT TO REALIZE THAT   002740     DBG=IDP    5846    A
DBG=IDP  *         THIS CODE, AS IT EXISTS IN /DBG=IDP/, IS A DATA SECTION, 002750 DBG=IDP    5847    A
DBG=IDP  *         I.E. IT CANNOT BE EXECUTED.  IT IS HERE ONLY SO THAT *FRZ* 002760 DBG=IDP    5848    A
DBG=IDP  *         CAN WRITE IT OUT AS THE 1ST RECORD ON *F.FRZ*.      002770     DBG=IDP    5849    A
DBG=IDP  *                                                             002780     DBG=IDP    5850    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *         ENTRY  NONE                                                  002790  DBG=IDP  5851  A
DBG=IDP  *                                                                      002800  DBG=IDP  5852  A
DBG=IDP  *         EXIT   TO *RHH* IN HIGH CORE TO READ IN FROZEN HOST          002810  DBG=IDP  5853  A
DBG=IDP  *                                                                      002820  DBG=IDP  5854  A
DBG=IDP  *         USES   IRRELEVANT                                            002830  DBG=IDP  5855  A
DBG=IDP  *                                                                      002840  DBG=IDP  5856  A
DBG=IDP  *         CALLS  MEMORY                                                002850  DBG=IDP  5857  A
DBG=IDP                                                                         002860  DBG=IDP  5858  A
DBG=IDP                                                                         002870  DBG=IDP  5859  A
DBG=IDP            LOC    RA.ORG                                                002880  DBG=IDP  5860  A
DBG=IDP                                                                         002890  DBG=IDP  5861  A
DBG=IDP            VFD    12/5000B,12/0,18/RA.ORG,18/IFR                        002900  DBG=IDP  5862  A
DBG=IDP                                                                         002910  DBG=IDP  5863  A
DBG=IDP   IFR      BSS    0           ** LOADER ENTRY POINT **                  002920  DBG=IDP  5864  A
DBG=IDP            SB1    1                                                     002930  DBG=IDP  5865  A
DBG=IDP                                                                         002940  DBG=IDP  5866  A
DBG=IDP  *         REQUEST ENOUGH MEMORY FOR THE FROZEN HOST PLUS *RHH*.        002950  DBG=IDP  5867  A
DBG=IDP  *         ALSO, IF *CMM* WAS ACTIVE IN FROZEN HOST, INDICATE SO TO     002960  DBG=IDP  5868  A
DBG=IDP  *         OPERATING SYSTEM.                                            002970  DBG=IDP  5869  A
DBG=IDP                                                                         002980  DBG=IDP  5870  A
DBG=IDP            SA2    MEM#HOST    (X2) = 30/AFLS OF FROZEN HOST,30/OTHER    002990  DBG=IDP  5871  A
DBG=IDP            SX6    L.RHH+L.IFRSZ+10B                                     003000  DBG=IDP  5872  A
DBG=IDP            LX6    30                                                    003010  DBG=IDP  5873  A
DBG=IDP            IX6    X2+X6       (X6) = 30/ENOUGH CORE FOR FROZEN HOST     003020  DBG=IDP  5874  A
DBG=IDP  *                                   AND *RHH*, 30/OTHER                003030  DBG=IDP  5875  A
DBG=IDP            SA6    MEM#RHH                                               003040  DBG=IDP  5876  A
DBG=IDP                                                                         003050  DBG=IDP  5877  A
DBG=IDP                                                                         003060  DBG=IDP  5878  A
DBG=IDP            SA1    IFRA        (X1) = *MEM* REQUEST WORD                 003070  DBG=IDP  5879  A
DBG=IDP            BX6    X1                                                    003080  DBG=IDP  5880  A
DBG=IDP            RJ     SYS         MAKE SYSTEM REQUEST                       003090  DBG=IDP  5881  A
DBG=IDP                                                                         003100  DBG=IDP  5882  A
DBG=IDP  *         SET UP *F.FRZ* FET SO THAT A *READSKP* WILL READ             003110  DBG=IDP  5883  A
DBG=IDP  *         THE FROZEN HOST FROM *F.FRZ/RECORD 2* INTO                   003120  DBG=IDP  5884  A
DBG=IDP  *         THE HOLE CREATED BY *MEM*.                                   003130  DBG=IDP  5885  A
DBG=IDP                                                                         003140  DBG=IDP  5886  A
DBG=IDP            SA1    RA.PGN                                                003150  DBG=IDP  5887  A
DBG=IDP            AX2    30                                                    003160  DBG=IDP  5888  A
DBG=IDP            SX7    X2          (X7) = OUT = AFLS OF FROZEN HOST          003170  DBG=IDP  5889  A
DBG=IDP            MX3    60-18                                                 003180  DBG=IDP  5890  A
DBG=IDP            SA7    F.FRZ+3     OUT = AFLS OF FROZEN HOST                 003190  DBG=IDP  5891  A
DBG=IDP            SX4    122B        (X4) = *OPEN/NO REWIND* CIO CODE          003200  DBG=IDP  5892  A
DBG=IDP            BX6    X3*X1                                                 003210  DBG=IDP  5893  A
DBG=IDP            SX7    X7+B1       (X7) = LIMIT = AFLS+1 OF FROZEN HOST      003220  DBG=IDP  5894  A
DBG=IDP            IX6    X6+X4                                                 003230  DBG=IDP  5895  A
DBG=IDP            SA7    A7+B1       LIMIT = AFLS+1 OF FROZEN HOST             003240  DBG=IDP  5896  A
DBG=IDP            SA6    F.FRZ                                                 003250  DBG=IDP  5897  A
DBG=IDP            =X7    RA.MTR+1                                              003260  DBG=IDP  5898  A
DBG=IDP            SA7    A6+B1       FIRST = RA.MTR+1                          003270  DBG=IDP  5899  A
DBG=IDP            SA7    A7+B1       IN = FIRST =RA.MTR+1                      003280  DBG=IDP  5900  A
DBG=IDP                                                                         003290  DBG=IDP  5901  A
DBG=IDP  *         MOVE *RHH* AND *F.FRZ* FET TO HIGH CORE                      003300  DBG=IDP  5902  A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP  *         (ABOVE HOLE CREATED FOR FROZEN HOST).                    003310  DBG=IDP  5903  A
DBG=IDP                                                                     003320  DBG=IDP  5904  A
DBG=IDP            SB4    B0                                                003330  DBG=IDP  5905  A
DBG=IDP            SA1    FW.RHH      (A1,X1) = A+C OF 1ST WORD TO MOVE     003340  DBG=IDP  5906  A
DBG=IDP            SB3    L.RHH       (B2) = NR OF WORDS TO MOVE            003350  DBG=IDP  5907  A
DBG=IDP            SB2    X2+L.IFRSZ  (B2) = ADDR TO MOVE TO                003360  DBG=IDP  5908  A
DBG=IDP                                                                     003370  DBG=IDP  5909  A
DBG=IDP   IFR2     BX6    X1                                                003380  DBG=IDP  5910  A
DBG=IDP            SA6    B2+B4                                             003390  DBG=IDP  5911  A
DBG=IDP            SB4    B4+B1                                             003400  DBG=IDP  5912  A
DBG=IDP            SA1    A1+B1                                             003410  DBG=IDP  5913  A
DBG=IDP            LT     B4,B3,IFR2  IF NOT DONE                           003420  DBG=IDP  5914  A
DBG=IDP                                                                     003430  DBG=IDP  5915  A
DBG=IDP  *         OPEN *F.FRZ*.                                            003440  DBG=IDP  5916  A
DBG=IDP                                                                     003450  DBG=IDP  5917  A
DBG=IDP            SA1    IFRB        (X1) = *CIO* REQUEST WORD             003460  DBG=IDP  5918  A
DBG=IDP            SX2    B2+F.FRZ-FW.RHH   (X2) = FWA OF *F.FRZ* FET       003470  DBG=IDP  5919  A
DBG=IDP            BX6    X1+X2                                             003480  DBG=IDP  5920  A
DBG=IDP            SA6    A1          SAVE *CIO* REQUEST W/ ADDR OF MOVED *F.FRZ*003490  DBG=IDP  5921  A
DBG=IDP            RJ     SYS         MAKE SYSTEM REQUEST TO OPEN *F.FRZ*   003500  DBG=IDP  5922  A
DBG=IDP                                                                     003510  DBG=IDP  5923  A
DBG=IDP  *         SET UP SYSTEM REQUEST TO PERFORM THE *READSKP*           003520  DBG=IDP  5924  A
DBG=IDP  *         WHICH WILL READ THE FROZEN HOST INTO THE HOLE CREATED    003530  DBG=IDP  5925  A
DBG=IDP  *         BY *MEM*.                                                003540  DBG=IDP  5926  A
DBG=IDP                                                                     003550  DBG=IDP  5927  A
DBG=IDP   IFR3     SA1    RA.MTR                                            003560  DBG=IDP  5928  A
DBG=IDP            NZ     X1,IFR3     IF (RA.MTR) NOT CLEAR, WAIT...        003570  DBG=IDP  5929  A
DBG=IDP                                                                     003580  DBG=IDP  5930  A
DBG=IDP            SA1    SYS1                                              003590  DBG=IDP  5931  A
DBG=IDP            SA2    B2+F.FRZ-FW.RHH   (A2,X2) = A+C OF *F.FRZ* FET WORD 1 003600  DBG=IDP  5932  A
DBG=IDP            SA3    IFRB        (X3) = *CIO* REQUEST WORD             003610  DBG=IDP  5933  A
DBG=IDP            MX4    60-18                                             003620  DBG=IDP  5934  A
DBG=IDP            BX7    X1                                                003630  DBG=IDP  5935  A
DBG=IDP            LX1    59-56       (X1) = .MI. IF *EQ* INSTRUCTION       003640  DBG=IDP  5936  A
DBG=IDP  *                                 = .PL. IF *XJ* INSTRUCTION       003650  DBG=IDP  5937  A
DBG=IDP            BX6    X4*X2       CLEAR BITS 0 THRU 17 OF *F.FRZ* WORD 1 003660  DBG=IDP  5938  A
DBG=IDP            SX4    22B         (X4) = *READSKP* CIO CODE             003670  DBG=IDP  5939  A
DBG=IDP            IX6    X6+X4       MERGE *READSKP* CIO CODE              003680  DBG=IDP  5940  A
DBG=IDP            SA6    A2                                                003690  DBG=IDP  5941  A
DBG=IDP            BX6    X3          (X6) = *CIO* REQUEST WORD             003700  DBG=IDP  5942  A
DBG=IDP                                                                     003710  DBG=IDP  5943  A
DBG=IDP  *         RELOCATE *WAIT (RA.MTR) LOOP* IF NOT USING               003720  DBG=IDP  5944  A
DBG=IDP  *         *XJ* INSTRUCTION.                                        003730  DBG=IDP  5945  A
DBG=IDP                                                                     003740  DBG=IDP  5946  A
DBG=IDP            PL     X1,IFR4     IF USING *XJ*                         003750  DBG=IDP  5947  A
DBG=IDP            MX2    -18                                               003760  DBG=IDP  5948  A
DBG=IDP            SX3    B2+B1       (X3) = ADDR OF *RHH* INSTRUCTIONS WHICH 003770  DBG=IDP  5949  A
DBG=IDP  *                                   ACTUALLY MAKES THE *CIO* REQUEST 003780  DBG=IDP  5950  A
DBG=IDP  *                                   TO READ *F.FRZ*                 003790  DBG=IDP  5951  A
DBG=IDP            BX7    X2*X7       CLEAR OLD ADDR                        003800  DBG=IDP  5952  A
DBG=IDP            IX7    X7+X3       MERGE NEW ADDR                        003810  DBG=IDP  5953  A
DBG=IDP            SA7    B2+B1                                             003820  DBG=IDP  5954  A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP                                                                  003830   DBG=IDP     5955     A
DBG=IDP  *         EVERYTHING IS READY TO GO...                          003840   DBG=IDP     5956     A
DBG=IDP  *         TRANSFER CONTROL TO *RHH* UP IN HIGH CORE.            003850   DBG=IDP     5957     A
DBG=IDP                                                                  003860   DBG=IDP     5958     A
DBG=IDP   IFR4     JP     B2+                                            003870   DBG=IDP     5959     A
DBG=IDP                                                                  003880   DBG=IDP     5960     A
DBG=IDP                                                                  003890   DBG=IDP     5961     A
DBG=IDP  *         *MEM* REQUEST WORD.                                   003900   DBG=IDP     5962     A
DBG=IDP                                                                  003910   DBG=IDP     5963     A
DBG=IDP   IFRA     VFD    18/3LMEM,3/2,21/0,18/MEM#RHH                   003920   DBG=IDP     5964     A
DBG=IDP                                                                  003930   DBG=IDP     5965     A
DBG=IDP  *         *CIO* REQUEST WORD.                                   003940   DBG=IDP     5966     A
DBG=IDP                                                                  003950   DBG=IDP     5967     A
DBG=IDP   IFRB     VFD    18/3LCIO,3/2,21/0,18/0                         003960   DBG=IDP     5968     A
DBG=IDP   SYS      SPACE  4                                              003970   DBG=IDP     5969     A
DBG=IDP  ***       SYS - PROCESS SYSTEM REQUEST.                         003980   DBG=IDP     5970     A
DBG=IDP  *                                                               003990   DBG=IDP     5971     A
DBG=IDP  *         ENTRY  (X6) = SYSTEM REQUEST.                         004000   DBG=IDP     5972     A
DBG=IDP  *                                                               004010   DBG=IDP     5973     A
DBG=IDP  *         EXIT   REQUEST PROCESSED.                             004020   DBG=IDP     5974     A
DBG=IDP  *                                                               004030   DBG=IDP     5975     A
DBG=IDP  *         USES   X - 1.                                         004040   DBG=IDP     5976     A
DBG=IDP  *                B - NONE.                                      004050   DBG=IDP     5977     A
DBG=IDP  *                A - 1, 6.                                      004060   DBG=IDP     5978     A
DBG=IDP  *                                                               004070   DBG=IDP     5979     A
DBG=IDP  *         CALLS  NONE.                                          004080   DBG=IDP     5980     A
DBG=IDP                                                                  004090   DBG=IDP     5981     A
DBG=IDP                                                                  004100   DBG=IDP     5982     A
DBG=IDP  *         INSTRUCTION WORDS FOR MONITOR CALL.                   004110   DBG=IDP     5983     A
DBG=IDP                                                                  004120   DBG=IDP     5984     A
DBG=IDP   SYSA     BSS    0                                              004130   DBG=IDP     5985     A
DBG=IDP            LOC    *+2                                            004140   DBG=IDP     5986     A
DBG=IDP   +        SA1    A1          WAIT (RA+1) CLEAR IF AUTO RECALL   004150   DBG=IDP     5987     A
DBG=IDP            LX1    59-40                                          004160   DBG=IDP     5988     A
DBG=IDP            NG     X1,*                                           004170   DBG=IDP     5989     A
DBG=IDP            LOC    SYSA+1                                         004180   DBG=IDP     5990     A
DBG=IDP                                                                  004190   DBG=IDP     5991     A
DBG=IDP            XJ                                                    004200   DBG=IDP     5992     A
DBG=IDP                                                                  004210   DBG=IDP     5993     A
DBG=IDP   SYS1     EQ     SYS2        FIRST ENTRY                        004220   DBG=IDP     5994     A
DBG=IDP                                                                  004230   DBG=IDP     5995     A
DBG=IDP   SYS      PS                 ENTRY/EXIT                         004240   DBG=IDP     5996     A
DBG=IDP   +        SA1    1           WAIT (RA+1) CLEAR                  004250   DBG=IDP     5997     A
DBG=IDP            NZ     X1,*                                           004260   DBG=IDP     5998     A
DBG=IDP            SA6    A1          ENTER REQUEST                      004270   DBG=IDP     5999     A
DBG=IDP            EQ     SYS1                                           004280   DBG=IDP     6000     A
DBG=IDP                                                                  004290   DBG=IDP     6001     A
DBG=IDP  *         INITIAL ENTRY TO SET TYPE OF CALL.                    004300   DBG=IDP     6002     A
DBG=IDP                                                                  004310   DBG=IDP     6003     A
DBG=IDP   SYS2     SA1    SYSA        NO MEJ/CEJ OPTION                  004320   DBG=IDP     6004     A
DBG=IDP            BX6    X1                                             004330   DBG=IDP     6005     A
DBG=IDP                                                                  004340   DBG=IDP     6006     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

```
DBG=IDP            SA1    RA.CEJ                                          004350   DBG=IDP     6007     A
DBG=IDP            PL     X1,SYS3     IF NO MEJ/CEJ PRESENT               004360   DBG=IDP     6008     A
DBG=IDP            SA1    SYSA+1                                          004370   DBG=IDP     6009     A
DBG=IDP            BX6    X1                                              004380   DBG=IDP     6010     A
DBG=IDP   SYS3     BSS    0                                               004390   DBG=IDP     6011     A
DBG=IDP                                                                   004400   DBG=IDP     6012     A
DBG=IDP            SA6    SYS1        SET MONITOR CALL                    004410   DBG=IDP     6013     A
DBG=IDP            SA1    1           RESET (A1)                          004420   DBG=IDP     6014     A
DBG=IDP            BX6    X1                                              004430   DBG=IDP     6015     A
DBG=IDP            RJ     SYS1-1      CLEAR STACK                         004440   DBG=IDP     6016     A
DBG=IDP   IFR      SPACE  4,8                                             004450   DBG=IDP     6017     A
DBG=IDP   MEM#RHH  BSSZ   1           30/AFLS OF HOST PLUS *RHH*,30/OTHER 004460   DBG=IDP     6018     A
DBG=IDP                                                                   004470   DBG=IDP     6019     A
DBG=IDP   MEM#HOST BSSZ   1           30/AFLS OF HOST,30/OTHER            004480   DBG=IDP     6020     A
DBG=IDP                                                                   004490   DBG=IDP     6021     A
DBG=IDP   L.IFRSZ  =      10B         LEN OF SAFETY ZONE BETWEEN HOST AND *RHH* 004500   DBG=IDP     6022     A
DBG=IDP   RHH      SPACE  4,8                                             004510   DBG=IDP     6023     A
DBG=IDP  **        RHH - READ HOST INTO HOLE.                             004520   DBG=IDP     6024     A
DBG=IDP  *                                                                004530   DBG=IDP     6025     A
DBG=IDP  *                                                                004540   DBG=IDP     6026     A
DBG=IDP  *         THIS ROUTINE MAKES THE ACTUAL OPERATING SYSTEM REQUEST 004550   DBG=IDP     6027     A
DBG=IDP  *         TO READ THE FROZEN HOST FROM *F.FRZ/RECORD 2*          004560   DBG=IDP     6028     A
DBG=IDP  *         INTO THE HOLE CREATED BY *IFR*.                        004570   DBG=IDP     6029     A
DBG=IDP  *                                                                004580   DBG=IDP     6030     A
DBG=IDP  *         ENTRY  (X6) = *CIO* REQUEST WORD                       004590   DBG=IDP     6031     A
DBG=IDP  *                                                                004600   DBG=IDP     6032     A
DBG=IDP  *         EXIT   TO *IDP=IFR* IN UNFROZEN HOST                   004610   DBG=IDP     6033     A
DBG=IDP  *                                                                004620   DBG=IDP     6034     A
DBG=IDP  *         USES                                                   004630   DBG=IDP     6035     A
DBG=IDP  *                                                                004640   DBG=IDP     6036     A
DBG=IDP  *         CALLS  NONE                                            004650   DBG=IDP     6037     A
DBG=IDP                                                                   004660   DBG=IDP     6038     A
DBG=IDP   FW.RHH   BSS    0           MARK FWA OF*RHH*                    004670   DBG=IDP     6039     A
DBG=IDP                                                                   004680   DBG=IDP     6040     A
DBG=IDP   RHH      BSS    0           ** ENTRY **                         004690   DBG=IDP     6041     A
DBG=IDP            SA6    RA.MTR                                          004700   DBG=IDP     6042     A
DBG=IDP                                                                   004710   DBG=IDP     6043     A
DBG=IDP  *         THE FOLLOWING LOCATION WILL BE PLUGGED WITH            004720   DBG=IDP     6044     A
DBG=IDP  *         A *WAIT (RA.MTR)* LOOP BY *IFR* IF NO *XJ* INSTRUCTION 004730   DBG=IDP     6045     A
DBG=IDP  *         AVAILABLE.                                             004740   DBG=IDP     6046     A
DBG=IDP                                                                   004750   DBG=IDP     6047     A
DBG=IDP  +         XJ                                                     004760   DBG=IDP     6048     A
DBG=IDP                                                                   004770   DBG=IDP     6049     A
DBG=IDP  *         THE INTERACTIVE HOST IS NOW BACK IN CORE... RE-ENTER *IDP*. 004780   DBG=IDP     6050     A
DBG=IDP                                                                   004790   DBG=IDP     6051     A
DBG=IDP  +         EQ     /"IDP"/IDP=IFR                                  004800   DBG=IDP     6052     A
DBG=IDP   F.FRZ    SPACE  4,8                                             004810   DBG=IDP     6053     A
DBG=IDP  **        F.FRZ - FET FOR *FREEZE* FILE.                         004820   DBG=IDP     6054     A
DBG=IDP                                                                   004830   DBG=IDP     6055     A
DBG=IDP   F.FRZ    BSS    0           ** FWA OF FET **                    004840   DBG=IDP     6056     A
DBG=IDP   L.FRZ    =      101B        LENGTH OF *F.FRZ* BUFFER FOR RECORD 1 WRITE004850   DBG=IDP     6057     A
DBG=IDP  *                            (I.E. SMALLEST POSSIBLE, 1 PRU)     004860   DBG=IDP     6058     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DBG=IDP

DBG=IDP   FRZ      FILEB  /"IDP"/FW.IFR,L.FRZ                          004870   DBG=IDP     6059     A
DBG=IDP                                                                004880   DBG=IDP     6060     A
DBG=IDP                                                                004890   DBG=IDP     6061     A
DBG=IDP   L.RHH    =      *-FW.RHH+1                                   004900   DBG=IDP     6062     A
DBG=IDP                                                                004910   DBG=IDP     6063     A
DBG=IDP            LOC    *O                                           004920   DBG=IDP     6064     A
DBG=IDP            QUAL   *                                            004930   DBG=IDP     6065     A
DBG=IDP                                                                004940   DBG=IDP     6066     A
DBG=IDP   L.IFR    =      *-FW.IFR+1  LENGTH OF IDP FREEZE RESTART PROGRAM  004950   DBG=IDP     6067     A
DBG=IDP   IFR      SPACE  4,8                                          004960   DBG=IDP     6068     A
DBG=IDP  **        REDEFINE SYMBOLS THAT *IDP* NEEDS TO REFERENCE INSIDE /IFR/.  004970   DBG=IDP     6069     A
DBG=IDP                                                                004980   DBG=IDP     6070     A
DBG=IDP   F.FRZ    =      FW.IFR+/IFR/F.FRZ-RA.ORG                     004990   DBG=IDP     6071     A
DBG=IDP   MEM#RHH  =      FW.IFR+/IFR/MEM#RHH-RA.ORG                   005000   DBG=IDP     6072     A
DBG=IDP   MEM#HOST =      FW.IFR+/IFR/MEM#HOST-RA.ORG                  005010   DBG=IDP     6073     A
DBG=IDP   IDP      SPACE  4,8                                          005020   DBG=IDP     6074     A
DBG=IDP                                                                         DBG=IDP     6075     A
DBG=IDP   #OS      ELSE                                                         DBG=IDP     6076     A
DBG=IDP   IDP=     SUBR              ** ENTRY/EXIT **                           DBG=IDP     6077     A
DBG=IDP            EQ     EXIT.                                                 DBG=IDP     6078     A
DBG=IDP   #OS      ENDIF                                                        DBG=IDP     6079     A
DBG=IDP   DBG=IDP  SPACE  4,8                                                   DBG=IDP     6080     A
DBG=IDP            BASE   *                                                     DBG=IDP     6081     A
DBG=IDP   QUAL$    IF     -DEF,QUAL$                                            DBG=IDP     6082     A
DBG=IDP            QUAL   *                                                     DBG=IDP     6083     A
DBG=IDP   IDP=     =      /DBG=IDP/IDP=                                         DBG=IDP     6084     A
DBG=IDP                                                                         DBG=IDP     6085     A
DBG=IDP            IF     DEF,/DBG=IDP/.OS,1                                    DBG=IDP     6086     A
DBG=IDP   .OS      =      /DBG=IDP/.OS                                          DBG=IDP     6087     A
DBG=IDP   #OS      IFNE   .OS,2       IF NOT SCOPE 2                           DBG=IDP     6088     A
DBG=IDP   IDP=ER   =      /DBG=IDP/IDP=ER                                       DBG=IDP     6089     A
DBG=IDP   IDP=MN   =      /DBG=IDP/IDP=MN                                       DBG=IDP     6090     A
DBG=IDP   #OS      ENDIF                                                        DBG=IDP     6091     A
DBG=IDP                                                                         DBG=IDP     6092     A
DBG=IDP   REG=     =      /DBG=IDP/REG=                                         DBG=IDP     6093     A
DBG=IDP   SNP=     =      /DBG=IDP/SNP=                                         DBG=IDP     6094     A
DBG=IDP   QUAL$    ENDIF                                                        DBG=IDP     6095     A
DBG=IDP            ENDX                                                         DBG=IDP     6096     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - DBG=IDP

          IDENTIFIER                TOTAL    ACTIVE

          DBG=IDP                    6096      6096

          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=CLO

FA=CLO    *COMDECK,FA=CLO                                                       FA=CLO         1     A
FA=CLO             CTEXT  FA=CLO - CLOSE FILE                                   FA=CLO         2     A
FA=CLO     CLO     SPACE  4                                                     FA=CLO         3     A
FA=CLO             IF     -DEF,QUAL$,1                                          FA=CLO         4     A
FA=CLO             QUAL   FA=CLO                                                FA=CLO         5     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=CLO

```
FA=CLO                                                                                  FA=CLO        6     A
FA=CLO                                                                                  FA=CLO        7     A
FA=CLO   **        FA=CLO - CLOSE FILE.                                                 FA=CLO        8     A
FA=CLO   *                                                                              FA=CLO        9     A
FA=CLO   *                                                                              FA=CLO       10     A
FA=CLO   *         CLOSES A FILE IF OPEN.                                               FA=CLO       11     A
FA=CLO   *                                                                              FA=CLO       12     A
FA=CLO   *         ENTRY  (X2) = FIT ADDRESS                                            FA=CLO       13     A
FA=CLO   *                                                                              FA=CLO       14     A
FA=CLO   *         EXIT   (X2) = FIT ADDRESS                                            FA=CLO       15     A
FA=CLO   *                                                                              FA=CLO       16     A
FA=CLO   *         USES   X - 3, 6                                                      FA=CLO       17     A
FA=CLO   *                A - 3                                                         FA=CLO       18     A
FA=CLO   *                B - NONE                                                      FA=CLO       19     A
FA=CLO   *                                                                              FA=CLO       20     A
FA=CLO   *         CALLS CLOSEM, FETCH                                                  FA=CLO       21     A
FA=CLO                                                                                  FA=CLO       22     A
FA=CLO                                                                                  FA=CLO       23     A
FA=CLO    FA=CLO   JP     *+4S15      ** ENTRY/EXIT **                                  FA=CLO       24     A
FA=CLO             FETCH  X2,OC,X6,X3                                                   FA=CLO       25     A
FA=CLO             SX6    X6-#OPE#                                                      FA=CLO       26     A
FA=CLO             NZ     X6,FA=CLO   IF NOT OPEN, EXIT                                 FA=CLO       27     A
FA=CLO             CLOSEM X2                                                            FA=CLO       28     A
FA=CLO             JP     FA=CLO      EXIT                                              FA=CLO       29     A
FA=CLO                                                                                  FA=CLO       30     A
FA=CLO                                                                                  FA=CLO       31     A
FA=CLO             IF     -DEF,QUAL$,2                                                  FA=CLO       32     A
FA=CLO             QUAL   *                                                             FA=CLO       33     A
FA=CLO    FA=CLO   EQU    /FA=CLO/FA=CLO                                                FA=CLO       34     A
FA=CLO             IF     -ABS,FA=CLO,1                                                 FA=CLO       35     A
FA=CLO             ENTRY  FA=CLO                                                        FA=CLO       36     A
FA=CLO                                                                                  FA=CLO       37     A
FA=CLO             ENDX                                                                 FA=CLO       38     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - FA=CLO

         IDENTIFIER                TOTAL    ACTIVE

         FA=CLO                       38        38

         LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=DEFS

FA=DEFS  *COMDECK,FA=DEFS                                                               FA=DEFS       1     A
FA=DEFS   FA=DEFS  TITLE  I/O CODING CONVENTIONS                                        FA=DEFS       2     A
FA=DEFS  **        I/O MACRO CODING CONVENTIONS.                                        FA=DEFS       3     A
FA=DEFS  *                                                                              FA=DEFS       4     A
FA=DEFS  *         1.  MACRO DEFINITIONS THAT ACCEPT NULL CALLING PARAMETERS            FA=DEFS       5     A
FA=DEFS  *         SHOULD BE AVOIDED, PARTICULARLY WHEN A NULL PARAMETER IMPLIES        FA=DEFS       6     A
FA=DEFS  *         THAT A REGISTER IS ALREADY PROPERLY LOADED.  THE ASSUMPTIONS         FA=DEFS       7     A
FA=DEFS  *         OR REGISTERS MAY HAVE TO BE CHANGED IN THE FUTURE, WHICH COULD       FA=DEFS       8     A
FA=DEFS  *         COMPEL RECODING OF ALL CALLS TO THE MACRO IF NULL PARAMETERS         FA=DEFS       9     A
FA=DEFS  *         HAVE SPECIFIC MEANINGS.                                              FA=DEFS      10     A

         0        1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=DEFS

```
FA=DEFS  *            THE PREFERRED TECHNIQUE IS TO REQUIRE EXPLICIT CALL          FA=DEFS    11   A
FA=DEFS  *        PARAMETERS, AND THEN SQUEEZE OUT REDUNDANT CODE WITH THE         FA=DEFS    12   A
FA=DEFS  *        (R=) PSEUDO-OP OR OTHER METHOD.                                  FA=DEFS    13   A
FA=DEFS  *                                                                         FA=DEFS    14   A
FA=DEFS  *        2.  *USES* LISTS ONLY THE REGISTERS DESTROYED BY THE MACRO       FA=DEFS    15   A
FA=DEFS  *        CALL EXPANSION.  REGISTERS DESTROYED BY INNER MACRO OR           FA=DEFS    16   A
FA=DEFS  *        SUBROUTINE CALLS ARE LISTED IN THE DESCRIPTION OF THE CALLED     FA=DEFS    17   A
FA=DEFS  *        ITEM.                                                            FA=DEFS    18   A
FA=DEFS  *            THE ONLY EXCEPTION TO THIS IS THAT REGISTERS DESTROYED BY    FA=DEFS    19   A
FA=DEFS  *        THE (R=) PSEUDO-OP SHOULD BE LISTED IN THE *USES* BLOCK OF       FA=DEFS    20   A
FA=DEFS  *        THE CALLER.                                                      FA=DEFS    21   A
FA=DEFS  *                                                                         FA=DEFS    22   A
FA=DEFS  *        3.  BOTH MACRO AND SUBROUTINE CALLS ARE LISTED IN *CALLS*.       FA=DEFS    23   A
FA=DEFS  *                                                                         FA=DEFS    24   A
FA=DEFS  *        4.  CONDITIONAL ASSEMBLY PSEUDO-OPS (IF.., ELSE, SKIP, ENDIF)    FA=DEFS    25   A
FA=DEFS  *        ARE ALWAYS EITHER COUNT-CONTROLLED OR LABELED, TO AVOID MACRO    FA=DEFS    26   A
FA=DEFS  *        NESTING PROBLEMS.                                                FA=DEFS    27   A
FA=DEFS  *            COUNT CONTROL IS PREFERRED WHERE THE LINE COUNTS ARE         FA=DEFS    28   A
FA=DEFS  *        SMALL (.LE. 9), TO REDUCE ASSEMBLY TIME.                         FA=DEFS    29   A
FA=DEFS  *            LABELING IS PREFERRED FOR LARGER LINE COUNTS, TO REDUCE      FA=DEFS    30   A
FA=DEFS  *        THE CHANCE OF COUNT ERRORS AND TO ENHANCE CODE READABILITY.      FA=DEFS    31   A
FA=DEFS  *                                                                         FA=DEFS    32   A
FA=DEFS  *        5.  ADDRESS TERMINOLOGY.                                         FA=DEFS    33   A
FA=DEFS  *               THE FOLLOWING CONVENTIONS ARE ADOPTED FOR REFERRING       FA=DEFS    34   A
FA=DEFS  *        TO A GROUP OF CONTIGUOUS STORAGE LOCATIONS.                      FA=DEFS    35   A
FA=DEFS  *            A.  *FWA* OR FIRST WORD ADDRESS IS THE SMALLEST STORAGE      FA=DEFS    36   A
FA=DEFS  *            ADDRESS FOR THE GROUP.                                       FA=DEFS    37   A
FA=DEFS  *            B.  *LWA* OR LAST WORD ADDRESS IS THE LARGEST STORAGE        FA=DEFS    38   A
FA=DEFS  *            ADDRESS FOR THE GROUP.                                       FA=DEFS    39   A
FA=DEFS  *            C.  *ADDRESS*, IF NOT OTHERWISE QUALIFIED, IS THE FIRST      FA=DEFS    40   A
FA=DEFS  *            WORD ADDRESS.                                                FA=DEFS    41   A
FA=DEFS  *            D. *FWA* IS PREFERRED WHEN BOTH *FWA* AND *LWA* ARE          FA=DEFS    42   A
FA=DEFS  *            BEING MENTIONED, TO AVOID CONFUSION.  OTHERWISE,             FA=DEFS    43   A
FA=DEFS  *            *ADDRESS* IS PREFERRED.                                      FA=DEFS    44   A
FA=DEFS  *            E.  *LWA+1* IS NEVER, NEVER ABBREVIATED AS *LWA*.            FA=DEFS    45   A
FA=DEFS           TITLE  I/O SYSTEM SYMBOLS                                        FA=DEFS    46   A
FA=DEFS           SPACE  4                                                         FA=DEFS    47   A
FA=DEFS   #RM     IFEQ   CP#RM,0                                                   FA=DEFS    48   A
FA=DEFS           SPACE  4,8                                                       FA=DEFS    49   A
FA=DEFS  ****     FET STRUCTURE AND WORD OFFSETS.                                  FA=DEFS    50   A
FA=DEFS  *                                                                         FA=DEFS    51   A
FA=DEFS  *                                                                         FA=DEFS    52   A
FA=DEFS  *            OFFSET        CONTENTS                                       FA=DEFS    53   A
FA=DEFS  *            ------        --------                                       FA=DEFS    54   A
FA=DEFS                                                                            FA=DEFS    55   A
FA=DEFS   I.FET    =      0          42/EXTERNAL LFN, 18/CIO CODE AND STATUS      FA=DEFS    56   A
FA=DEFS   I.FIRST  =      1          36/0, 6/FET LEN - 5, 18/FIRST                FA=DEFS    57   A
FA=DEFS   I.DT     =      1          12/DEVICE TYPE,48/0                          FA=DEFS    58   A
FA=DEFS   I.IN     =      2          60/IN                                        FA=DEFS    59   A
FA=DEFS   I.OUT    =      3          60/OUT                                       FA=DEFS    60   A
FA=DEFS   I.LIMIT  =      4          42/0,18/LIMIT                                FA=DEFS    61   A
FA=DEFS   I.PRU    =      4          26/0,16/PRU SIZE,18/0                        FA=DEFS    62   A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=DEFS

```
FA=DEFS   I.CBSET  =      5          24/0, 18/PASS 1 LEN, 18/PASS 1 FWA          FA=DEFS    63    A
FA=DEFS   I.RAN1   =      6          RANDOM FILE INFORMATION                     FA=DEFS    64    A
FA=DEFS   I.RAN2   =      7          RANDOM FILE INFORMATION                     FA=DEFS    65    A
FA=DEFS  ****                                                                    FA=DEFS    66    A
FA=DEFS            SPACE  4                                                      FA=DEFS    67    A
FA=DEFS   #RM      ELSE                                                          FA=DEFS    68    A
FA=DEFS            SPACE  4                                                      FA=DEFS    69    A
FA=DEFS  ****      PSEUDO-FET STRUCTURE AND WORD OFFSETS.                        FA=DEFS    70    A
FA=DEFS  *                                                                       FA=DEFS    71    A
FA=DEFS  *                                                                       FA=DEFS    72    A
FA=DEFS  *             OFFSET        CONTENTS                                    FA=DEFS    73    A
FA=DEFS  *             ------        --------                                    FA=DEFS    74    A
FA=DEFS                                                                          FA=DEFS    75    A
FA=DEFS   I.CBSET  =      -1         CIO BUFFER PASS 1 PARAMS - 42/LEN, 18/FWA   FA=DEFS    76    A
FA=DEFS   I.FIT    =      0          1/HOLD BUF FLAG, 1/SPILL FLAG, 58/FIT ADDR  FA=DEFS    77    A
FA=DEFS   I.FIRST  =      1          60/FIRST                                    FA=DEFS    78    A
FA=DEFS   I.IN     =      2          60/IN                                       FA=DEFS    79    A
FA=DEFS   I.OUT    =      3          60/OUT                                      FA=DEFS    80    A
FA=DEFS   I.LIMIT  =      4          60/LIMIT                                    FA=DEFS    81    A
FA=DEFS   I.HBSET  =      5          HOLDING BUF PASS 1 PARAMS - 42/LEN, 18/FWA  FA=DEFS    82    A
FA=DEFS  ****                                                                    FA=DEFS    83    A
FA=DEFS            SPACE  4                                                      FA=DEFS    84    A
FA=DEFS   #RM      ENDIF                                                         FA=DEFS    85    A
FA=DEFS            TITLE  I/O GENERAL MACROS                                     FA=DEFS    86    A
FA=DEFS   SETFIL   SPACE  4,8                                                    FA=DEFS    87    A
FA=DEFS  **        SETFIL - SET PARAMS IN FILE MANAGEMENT TABLES.                FA=DEFS    88    A
FA=DEFS  *                                                                       FA=DEFS    89    A
FA=DEFS  *                                                                       FA=DEFS    90    A
FA=DEFS  *         TO SET INITIAL PARAMETER VALUES IN FILE TABLES,               FA=DEFS    91    A
FA=DEFS  *         SETFIL    FILE=FNAM,MODE=INIT,FWA=ADDR                        FA=DEFS    92    A
FA=DEFS  *          OR                                                           FA=DEFS    93    A
FA=DEFS  *         SETFIL    FILE=FNAM,MODE=INIT,LWA1=ADDR                       FA=DEFS    94    A
FA=DEFS  *                                                                       FA=DEFS    95    A
FA=DEFS  *         TO RESTORE FILE TABLES TO INITIAL (PASS 1) CONDITIONS,        FA=DEFS    96    A
FA=DEFS  *         SETFIL    FILE=FNAM,MODE=RESET                                FA=DEFS    97    A
FA=DEFS  *                                                                       FA=DEFS    98    A
FA=DEFS  *         ENTRY  *FNAM* = FILE NAME                                     FA=DEFS    99    A
FA=DEFS  *                 MODE  = *INIT* OR *RESET*                             FA=DEFS   100    A
FA=DEFS  *                *ADDR* = BUFFER FWA OR LWA+1, TO MATCH EQUIV PARAM     FA=DEFS   101    A
FA=DEFS  *                                                                       FA=DEFS   102    A
FA=DEFS  *         USES   X - 1, 2                                               FA=DEFS   103    A
FA=DEFS  *                B - 2, 3                                               FA=DEFS   104    A
FA=DEFS  *                                                                       FA=DEFS   105    A
FA=DEFS  *         CALLS  FA=SET                                                 FA=DEFS   106    A
FA=DEFS                                                                          FA=DEFS   107    A
FA=DEFS                                                                          FA=DEFS   108    A
FA=DEFS            PURGMAC   SETFIL                                              FA=DEFS   109    A
FA=DEFS                                                                          FA=DEFS   110    A
FA=DEFS   SETFIL   MACROE FILE,MODE,FWA,LWA1                                     FA=DEFS   111    A
FA=DEFS            LOCAL  CHECK                                                  FA=DEFS   112    A
FA=DEFS   CHECK    SET    0                                                      FA=DEFS   113    A
FA=DEFS            IFC    NE, LWA1  ,1                                           FA=DEFS   114    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=DEFS

```
FA=DEFS   CHECK    SET    CHECK+1                                          FA=DEFS     115    A
FA=DEFS            IFC    NE, FWA  ,1                                      FA=DEFS     116    A
FA=DEFS   CHECK    SET    CHECK+2                                          FA=DEFS     117    A
FA=DEFS   *                          SET FILE ADDRESS.                    FA=DEFS     118    A
FA=DEFS            IFEQ   CP#RM,0,2                                        FA=DEFS     119    A
FA=DEFS            R=     X2,FILE                                          FA=DEFS     120    A
FA=DEFS            SKIP   1                                                FA=DEFS     121    A
FA=DEFS            R=     A2,FILE                                          FA=DEFS     122    A
FA=DEFS    .M      IFC    NE, MODE INIT                                    FA=DEFS     123    A
FA=DEFS   *                          VALIDATE MODE PARAMETER.             FA=DEFS     124    A
FA=DEFS            IFC    NE, MODE RESET ,1                                FA=DEFS     125    A
FA=DEFS            ERR    (MODE) BAD MODE# "SEQUENCE"                      FA=DEFS     126    A
FA=DEFS   *                          RESET MODE.                          FA=DEFS     127    A
FA=DEFS            IFNE   CHECK,0,1                                        FA=DEFS     128    A
FA=DEFS    7       ERR    MODE#RESET, FWA#/LWA1# IGNORED "SEQUENCE"        FA=DEFS     129    A
FA=DEFS            IFEQ   CP#RM,0,2                                        FA=DEFS     130    A
FA=DEFS            SA1    X2+I.CBSET                                       FA=DEFS     131    A
FA=DEFS            SKIP   1                                                FA=DEFS     132    A
FA=DEFS            SA1    A2+I.HBSET                                       FA=DEFS     133    A
FA=DEFS            SB3    B0                                               FA=DEFS     134    A
FA=DEFS            SB2    X1                                               FA=DEFS     135    A
FA=DEFS    .M      ELSE                                                    FA=DEFS     136    A
FA=DEFS   *                          INITIALIZE MODE.                     FA=DEFS     137    A
FA=DEFS            IFGE   CHECK,2,5                                        FA=DEFS     138    A
FA=DEFS            IFGE   CHECK,3,1                                        FA=DEFS     139    A
FA=DEFS    7       ERR    BOTH FWA# AND LWA1# SPECIFIED, LWA1# IGNORED "SEQUENCE"  FA=DEFS  140  A
FA=DEFS   *                          SET FWA.                             FA=DEFS     141    A
FA=DEFS            R=     B2,FWA                                           FA=DEFS     142    A
FA=DEFS            SB3    B0                                               FA=DEFS     143    A
FA=DEFS            SKIP   5                                                FA=DEFS     144    A
FA=DEFS            IFLE   CHECK,0,2                                        FA=DEFS     145    A
FA=DEFS            ERR    NO FWA#/LWA1# "SEQUENCE"                         FA=DEFS     146    A
FA=DEFS            SKIP   2                                                FA=DEFS     147    A
FA=DEFS   *                          SET LWA1.                            FA=DEFS     148    A
FA=DEFS            R=     B3,LWA1                                          FA=DEFS     149    A
FA=DEFS            SB2    B0                                               FA=DEFS     150    A
FA=DEFS    .M      ENDIF                                                   FA=DEFS     151    A
FA=DEFS   *                          CALL PROCESSING SUBROUTINE.          FA=DEFS     152    A
FA=DEFS            RJ     =XFA=SET                                         FA=DEFS     153    A
FA=DEFS   SETFIL   ENDM                                                    FA=DEFS     154    A
FA=DEFS            TITLE  FILE ACTION MACROS (RECORD MANAGER I/O)          FA=DEFS     155    A
FA=DEFS    #RM     IFNE   CP#RM,0                                          FA=DEFS     156    A
FA=DEFS    BKSP    SPACE  4,8                                              FA=DEFS     157    A
FA=DEFS   **       BKSP - BACKSPACE FILE ONE RECORD.                       FA=DEFS     158    A
FA=DEFS   *                                                                FA=DEFS     159    A
FA=DEFS   *                                                                FA=DEFS     160    A
FA=DEFS   *        BKSP      FILE                                          FA=DEFS     161    A
FA=DEFS   *                                                                FA=DEFS     162    A
FA=DEFS   *        ENTRY  *FILE* = FILE NAME                              FA=DEFS     163    A
FA=DEFS   *                                                                FA=DEFS     164    A
FA=DEFS   *        USES   X2, A2                                           FA=DEFS     165    A
FA=DEFS   *                                                                FA=DEFS     166    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=DEFS

```
FA=DEFS  *         CALLS  SKIPBL                                                FA=DEFS     167    A
FA=DEFS                                                                         FA=DEFS     168    A
FA=DEFS                                                                         FA=DEFS     169    A
FA=DEFS            PURGMAC   BKSP                                               FA=DEFS     170    A
FA=DEFS                                                                         FA=DEFS     171    A
FA=DEFS   BKSP     MACRO  F                                                     FA=DEFS     172    A
FA=DEFS            R=     A2,F                                                  FA=DEFS     173    A
FA=DEFS            SKIPBL X2,1                                                  FA=DEFS     174    A
FA=DEFS   BKSP     ENDM                                                         FA=DEFS     175    A
FA=DEFS   CLOSE    SPACE  4,8                                                   FA=DEFS     176    A
FA=DEFS  **        CLOSE - CLOSE FILE.                                          FA=DEFS     177    A
FA=DEFS  *                                                                      FA=DEFS     178    A
FA=DEFS  *         CLOSES *FILE*, USING POSITIONING PARAMETER (CF) CURRENTLY    FA=DEFS     179    A
FA=DEFS  *         IN THE FIT.                                                  FA=DEFS     180    A
FA=DEFS  *                                                                      FA=DEFS     181    A
FA=DEFS  *                                                                      FA=DEFS     182    A
FA=DEFS  *         CLOSE     FILE                                               FA=DEFS     183    A
FA=DEFS  *                                                                      FA=DEFS     184    A
FA=DEFS  *         ENTRY  *FILE* = FILE NAME                                    FA=DEFS     185    A
FA=DEFS  *                                                                      FA=DEFS     186    A
FA=DEFS  *         USES   X2, A2                                                FA=DEFS     187    A
FA=DEFS  *                                                                      FA=DEFS     188    A
FA=DEFS  *         CALLS  FA=CLO                                                FA=DEFS     189    A
FA=DEFS                                                                         FA=DEFS     190    A
FA=DEFS                                                                         FA=DEFS     191    A
FA=DEFS            PURGMAC   CLOSE                                              FA=DEFS     192    A
FA=DEFS                                                                         FA=DEFS     193    A
FA=DEFS   CLOSE    MACRO  F                                                     FA=DEFS     194    A
FA=DEFS            R=     A2,F                                                  FA=DEFS     195    A
FA=DEFS            RJ     =XFA=CLO                                              FA=DEFS     196    A
FA=DEFS   CLOSE    ENDM                                                         FA=DEFS     197    A
FA=DEFS   OPEN     SPACE  4,8                                                   FA=DEFS     198    A
FA=DEFS  **        OPEN - OPEN FILE.                                            FA=DEFS     199    A
FA=DEFS  *                                                                      FA=DEFS     200    A
FA=DEFS  *         OPENS *FILE*, USING PROCESSING DIRECTION (PD) AND POSITIONING FA=DEFS    201    A
FA=DEFS  *         (OF) PARAMETERS CURRENTLY IN THE FIT.                        FA=DEFS     202    A
FA=DEFS  *                                                                      FA=DEFS     203    A
FA=DEFS  *                                                                      FA=DEFS     204    A
FA=DEFS  *         OPEN      FILE                                               FA=DEFS     205    A
FA=DEFS  *                                                                      FA=DEFS     206    A
FA=DEFS  *         ENTRY  *FILE* = FILE NAME                                    FA=DEFS     207    A
FA=DEFS  *                                                                      FA=DEFS     208    A
FA=DEFS  *         USES   X2, A2                                                FA=DEFS     209    A
FA=DEFS  *                                                                      FA=DEFS     210    A
FA=DEFS  *         CALLS  FA=OPE                                                FA=DEFS     211    A
FA=DEFS                                                                         FA=DEFS     212    A
FA=DEFS                                                                         FA=DEFS     213    A
FA=DEFS            PURGMAC   OPEN                                               FA=DEFS     214    A
FA=DEFS                                                                         FA=DEFS     215    A
FA=DEFS   OPEN     MACRO  F                                                     FA=DEFS     216    A
FA=DEFS            R=     A2,F                                                  FA=DEFS     217    A
FA=DEFS            RJ     =XFA=OPE                                              FA=DEFS     218    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=DEFS

```
FA=DEFS   OPEN     ENDM                                                    FA=DEFS     219    A
FA=DEFS   READ     SPACE  3                                                FA=DEFS     220    A
FA=DEFS  *         EMPTY MACRO DEFINITIONS FOR READ AND RECALL FUNCTIONS   FA=DEFS     221    A
FA=DEFS  *         SO THE CONDITIONAL CODE STAYS IN THE TEXTS.             FA=DEFS     222    A
FA=DEFS                                                                    FA=DEFS     223    A
FA=DEFS            PURGMAC  READ,RECALL                                    FA=DEFS     224    A
FA=DEFS                                                                    FA=DEFS     225    A
FA=DEFS   READ     MACRO  F                                                FA=DEFS     226    A
FA=DEFS            ENDM                                                    FA=DEFS     227    A
FA=DEFS                                                                    FA=DEFS     228    A
FA=DEFS   RECALL   OPSYN  READ                                             FA=DEFS     229    A
FA=DEFS   REWIND   SPACE  3,8                                              FA=DEFS     230    A
FA=DEFS  **        REWIND- REWIND A FILE/BUFFER.                           FA=DEFS     231    A
FA=DEFS  *                                                                 FA=DEFS     232    A
FA=DEFS  *         THIS EXTENSION OF A NORMAL *REWIND* PREVENTS UNNECESSARY I/O  FA=DEFS  233  A
FA=DEFS  *         TRANSFERS OF SMALL INTERNAL SCRATCH FILES BETWEEN CENTRAL  FA=DEFS  234  A
FA=DEFS  *         MEMORY AND MASS STORAGE.                                FA=DEFS     235    A
FA=DEFS  *                                                                 FA=DEFS     236    A
FA=DEFS  *                                                                 FA=DEFS     237    A
FA=DEFS  *         REWIND    FILE,RECALL                                   FA=DEFS     238    A
FA=DEFS  *                                                                 FA=DEFS     239    A
FA=DEFS  *         ENTRY  *FILE*    = FILE NAME                           FA=DEFS     240    A
FA=DEFS  *               *RECALL*  = IGNORED; INCLUDED FOR CIO COMPATIBILITY  FA=DEFS  241  A
FA=DEFS  *                                                                 FA=DEFS     242    A
FA=DEFS  *         USES   X2, A2                                           FA=DEFS     243    A
FA=DEFS  *                                                                 FA=DEFS     244    A
FA=DEFS  *         CALLS  FA=RWX                                           FA=DEFS     245    A
FA=DEFS                                                                    FA=DEFS     246    A
FA=DEFS                                                                    FA=DEFS     247    A
FA=DEFS            PURGMAC   REWIND                                        FA=DEFS     248    A
FA=DEFS                                                                    FA=DEFS     249    A
FA=DEFS   REWIND   MACRO  F                                                FA=DEFS     250    A
FA=DEFS            R=     A2,F                                             FA=DEFS     251    A
FA=DEFS            RJ     =XFA=RWX                                         FA=DEFS     252    A
FA=DEFS   REWIND   ENDM                                                    FA=DEFS     253    A
FA=DEFS   WRITEF   SPACE  4,8                                              FA=DEFS     254    A
FA=DEFS  **        WRITEF - WRITE END-OF-FILE/PARTITION ON FILE.           FA=DEFS     255    A
FA=DEFS  *                                                                 FA=DEFS     256    A
FA=DEFS  *         WRITES END-OF-FILE/PARTITION ON *FILE*.  THE HOLDING BUFFER,  FA=DEFS  257  A
FA=DEFS  *         IF ONE EXISTS, IS FLUSHED BEFORE WRITING THE E-O-F.     FA=DEFS     258    A
FA=DEFS  *                                                                 FA=DEFS     259    A
FA=DEFS  *                                                                 FA=DEFS     260    A
FA=DEFS  *         WRITEF    FILE                                          FA=DEFS     261    A
FA=DEFS  *                                                                 FA=DEFS     262    A
FA=DEFS  *         ENTRY  *FILE* = FILE NAME                              FA=DEFS     263    A
FA=DEFS  *                                                                 FA=DEFS     264    A
FA=DEFS  *         USES   X2, A2                                           FA=DEFS     265    A
FA=DEFS  *                                                                 FA=DEFS     266    A
FA=DEFS  *         CALLS  FA=FLSH, RECORD MANAGER                          FA=DEFS     267    A
FA=DEFS                                                                    FA=DEFS     268    A
FA=DEFS                                                                    FA=DEFS     269    A
FA=DEFS            PURGMAC   WRITEF                                        FA=DEFS     270    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=DEFS

```
FA=DEFS                                                                  FA=DEFS     271    A
FA=DEFS   WRITEF   MACRO  F                                              FA=DEFS     272    A
FA=DEFS            R=     A2,F                                           FA=DEFS     273    A
FA=DEFS            RJ     =XFA=EOF                                       FA=DEFS     274    A
FA=DEFS   WRITEF   ENDM                                                  FA=DEFS     275    A
FA=DEFS   WRITER   SPACE  4,8                                            FA=DEFS     276    A
FA=DEFS  **        WRITER - WRITE END-OF-RECORD/SECTION ON FILE.         FA=DEFS     277    A
FA=DEFS  *                                                               FA=DEFS     278    A
FA=DEFS  *         WRITES LEVEL 0 END-OF-RECORD/SECTION ON *FILE*.  THE HOLDING  FA=DEFS  279  A
FA=DEFS  *         BUFFER, IF ONE EXISTS, IS FLUSHED BEFORE WRITING THE E-O-R.  FA=DEFS  280  A
FA=DEFS  *                                                               FA=DEFS     281    A
FA=DEFS  *                                                               FA=DEFS     282    A
FA=DEFS  *         WRITER    FILE                                       FA=DEFS     283    A
FA=DEFS  *                                                               FA=DEFS     284    A
FA=DEFS  *         ENTRY  *FILE* = FILE NAME                             FA=DEFS     285    A
FA=DEFS  *                                                               FA=DEFS     286    A
FA=DEFS  *         USES   X2, A2                                         FA=DEFS     287    A
FA=DEFS  *                                                               FA=DEFS     288    A
FA=DEFS  *         CALLS  FA=FLSH, RECORD MANAGER                        FA=DEFS     289    A
FA=DEFS                                                                  FA=DEFS     290    A
FA=DEFS                                                                  FA=DEFS     291    A
FA=DEFS            PURGMAC   WRITER                                      FA=DEFS     292    A
FA=DEFS                                                                  FA=DEFS     293    A
FA=DEFS   WRITER   MACRO  F                                              FA=DEFS     294    A
FA=DEFS            R=     A2,F                                           FA=DEFS     295    A
FA=DEFS            RJ     =XFA=EOR                                       FA=DEFS     296    A
FA=DEFS   WRITER   ENDM                                                  FA=DEFS     297    A
FA=DEFS   #RM      ENDIF                                                 FA=DEFS     298    A
FA=DEFS            TITLE  I/O DATA TRANSFER MACROS (RECORD MANAGER I/O)  FA=DEFS     299    A
FA=DEFS   #RM      IFNE   CP#RM,0                                        FA=DEFS     300    A
FA=DEFS   READC    SPACE  4,8                                            FA=DEFS     301    A
FA=DEFS  **        READC - READ ONE CODED LINE IN -C- FORMAT.            FA=DEFS     302    A
FA=DEFS  *                                                               FA=DEFS     303    A
FA=DEFS  *                                                               FA=DEFS     304    A
FA=DEFS  *         READC     FILE,TO,WORDS                               FA=DEFS     305    A
FA=DEFS  *                                                               FA=DEFS     306    A
FA=DEFS  *         ENTRY  *FILE*  = FILE NAME                            FA=DEFS     307    A
FA=DEFS  *                *TO*    = WSA ADDRESS                          FA=DEFS     308    A
FA=DEFS  *                *WORDS* = WSA LENGTH (WORDS)                   FA=DEFS     309    A
FA=DEFS  *                                                               FA=DEFS     310    A
FA=DEFS  *         USES   X2, A2, B6, B7                                 FA=DEFS     311    A
FA=DEFS  *                                                               FA=DEFS     312    A
FA=DEFS  *         CALLS  FA=RDC                                         FA=DEFS     313    A
FA=DEFS                                                                  FA=DEFS     314    A
FA=DEFS                                                                  FA=DEFS     315    A
FA=DEFS            PURGMAC   READC                                       FA=DEFS     316    A
FA=DEFS                                                                  FA=DEFS     317    A
FA=DEFS   READC    MACRO  F,S,N                                          FA=DEFS     318    A
FA=DEFS            R=     B6,S                                           FA=DEFS     319    A
FA=DEFS            R=     B7,N                                           FA=DEFS     320    A
FA=DEFS            R=     A2,F                                           FA=DEFS     321    A
FA=DEFS            RJ     =XFA=RDC                                       FA=DEFS     322    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=DEFS

```
FA=DEFS   READC    ENDM                                                    FA=DEFS      323     A
FA=DEFS   READW    SPACE  4,8                                              FA=DEFS      324     A
FA=DEFS  **        READW - READ WORDS TO WORKING STORAGE AREA.             FA=DEFS      325     A
FA=DEFS  *                                                                 FA=DEFS      326     A
FA=DEFS  *                                                                 FA=DEFS      327     A
FA=DEFS  *         READW     FILE,TO,WORDS                                 FA=DEFS      328     A
FA=DEFS  *                                                                 FA=DEFS      329     A
FA=DEFS  *         ENTRY  *FILE*  = FILE NAME                              FA=DEFS      330     A
FA=DEFS  *                *TO*    = WSA ADDRESS                            FA=DEFS      331     A
FA=DEFS  *                *WORDS* = WSA LENGTH (WORDS)                     FA=DEFS      332     A
FA=DEFS  *                                                                 FA=DEFS      333     A
FA=DEFS  *         USES   X2, A2, B6, B7                                   FA=DEFS      334     A
FA=DEFS  *                                                                 FA=DEFS      335     A
FA=DEFS  *         CALLS  FA=RDW                                           FA=DEFS      336     A
FA=DEFS                                                                    FA=DEFS      337     A
FA=DEFS                                                                    FA=DEFS      338     A
FA=DEFS            PURGMAC   READW                                         FA=DEFS      339     A
FA=DEFS                                                                    FA=DEFS      340     A
FA=DEFS   READW    MACRO  F,S,N                                            FA=DEFS      341     A
FA=DEFS            R=     B6,S                                             FA=DEFS      342     A
FA=DEFS            R=     B7,N                                             FA=DEFS      343     A
FA=DEFS            R=     A2,F                                             FA=DEFS      344     A
FA=DEFS            RJ     =XFA=RDW                                         FA=DEFS      345     A
FA=DEFS   READW    ENDM                                                    FA=DEFS      346     A
FA=DEFS   WRITEC   SPACE  4,8                                              FA=DEFS      347     A
FA=DEFS  **        WRITEC - WRITE ONE CODED LINE IN -C- FORMAT.            FA=DEFS      348     A
FA=DEFS  *                                                                 FA=DEFS      349     A
FA=DEFS  *                                                                 FA=DEFS      350     A
FA=DEFS  *         WRITEC    FILE,FROM,WORDS                               FA=DEFS      351     A
FA=DEFS  *                                                                 FA=DEFS      352     A
FA=DEFS  *         ENTRY  *FILE*  = FILE NAME                              FA=DEFS      353     A
FA=DEFS  *                *FROM*  = WSA ADDRESS                            FA=DEFS      354     A
FA=DEFS  *                *WORDS* = LINE LENGTH (WORDS).  IF ZERO OR OMITTED,  FA=DEFS  355     A
FA=DEFS  *                          LENGTH WILL BE DETERMINED BY SCANNING TO FA=DEFS      356     A
FA=DEFS  *                          AN END-OF-LINE MARKER -- SLOWLY.       FA=DEFS      357     A
FA=DEFS  *                                                                 FA=DEFS      358     A
FA=DEFS  *         USES   X2, A2, B6, B7                                   FA=DEFS      359     A
FA=DEFS  *                                                                 FA=DEFS      360     A
FA=DEFS  *         CALLS  FA=WTC                                           FA=DEFS      361     A
FA=DEFS                                                                    FA=DEFS      362     A
FA=DEFS                                                                    FA=DEFS      363     A
FA=DEFS            PURGMAC   WRITEC                                        FA=DEFS      364     A
FA=DEFS                                                                    FA=DEFS      365     A
FA=DEFS   WRITEC   MACRO  F,S,N                                            FA=DEFS      366     A
FA=DEFS            R=     B6,S                                             FA=DEFS      367     A
FA=DEFS            IFC    NE, N  ,2                                        FA=DEFS      368     A
FA=DEFS  *                          SET GIVEN LENGTH.                      FA=DEFS      369     A
FA=DEFS            R=     B7,N                                             FA=DEFS      370     A
FA=DEFS            SKIP   1                                                FA=DEFS      371     A
FA=DEFS  *                          SET LENGTH OMITTED.                    FA=DEFS      372     A
FA=DEFS            SB7    B0                                               FA=DEFS      373     A
FA=DEFS            R=     A2,F                                             FA=DEFS      374     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=DEFS

```
FA=DEFS            RJ     =XFA=WTC                                                FA=DEFS      375     A
FA=DEFS   WRITEC   ENDM                                                           FA=DEFS      376     A
FA=DEFS   WRITEH   SPACE  4,10                                                    CCGA018       41     A
FA=DEFS  **        WRITEH - WRITE ONE CODED LINE IN -H- FORMAT.                   CCGA018       42     A
FA=DEFS  *                                                                        CCGA018       43     A
FA=DEFS  *         WRITEH FILE,FROM,WORDS                                         CCGA018       44     A
FA=DEFS  *                                                                        CCGA018       45     A
FA=DEFS  *         ENTRY  *FILE* = FILE NAME                                      CCGA018       46     A
FA=DEFS  *                *FROM* = WSA ADDRESS                                    CCGA018       47     A
FA=DEFS  *                *WORDS* = LINE LENGTH (WORDS) MUST BE SPECIFIED.        CCGA018       48     A
FA=DEFS  *                                                                        CCGA018       49     A
FA=DEFS  *         THIS IS A *TEMPORARY* REDEFINITION OF THE SYSTEM WRITEH MACRO  CCGA018       50     A
FA=DEFS  *                                                                        CCGA018       51     A
FA=DEFS  *         USES   X2,A2,B3,B6,B7                                          CCGA018       52     A
FA=DEFS  *                                                                        CCGA018       53     A
FA=DEFS  *         CALLS  FA=WTH                                                  CCGA018       54     A
FA=DEFS                                                                           CCGA018       55     A
FA=DEFS                                                                           CCGA018       56     A
FA=DEFS            PURGMAC WRITEH                                                 CCGA018       57     A
FA=DEFS                                                                           CCGA018       58     A
FA=DEFS   WRITEH   MACRO  F,S,N                                                   CCGA018       59     A
FA=DEFS            SB3    X2          PRESERVE X2                                 CCGA018       60     A
FA=DEFS            R=     B6,S                                                   CCGA018       61     A
FA=DEFS            IFC    NE, N  ,2                                              CCGA018       62     A
FA=DEFS  *                            SET GIVEN LENGTH                            CCGA018       63     A
FA=DEFS            R=     B7,N                                                   CCGA018       64     A
FA=DEFS            SKIP   1                                                      CCGA018       65     A
FA=DEFS  *                            LENGTH OMITTED - ERROR                      CCGA018       66     A
FA=DEFS   A        ERR                                                            CCGA018       67     A
FA=DEFS            R=     A2,F                                                   CCGA018       68     A
FA=DEFS            RJ     =XFA=WTH                                               CCGA018       69     A
FA=DEFS            SX2    B3          RESTORE X2                                 CCGA018       70     A
FA=DEFS   WRITEH   ENDM                                                           CCGA018       71     A
FA=DEFS   WRITEW   SPACE  4,8                                                     FA=DEFS      377     A
FA=DEFS  **        WRITEW - WRITE WORDS FROM WORKING STORAGE AREA.                FA=DEFS      378     A
FA=DEFS  *                                                                        FA=DEFS      379     A
FA=DEFS  *                                                                        FA=DEFS      380     A
FA=DEFS  *         WRITEW    FILE,FROM,WORDS                                     FA=DEFS      381     A
FA=DEFS  *                                                                        FA=DEFS      382     A
FA=DEFS  *         ENTRY  *FILE*  = FILE NAME                                     FA=DEFS      383     A
FA=DEFS  *                *FROM*  = WSA ADDRESS                                   FA=DEFS      384     A
FA=DEFS  *                *WORDS* = WSA LENGTH (WORDS)                            FA=DEFS      385     A
FA=DEFS  *                                                                        FA=DEFS      386     A
FA=DEFS  *         USES   X2, A2, B6, B7                                          FA=DEFS      387     A
FA=DEFS  *                                                                        FA=DEFS      388     A
FA=DEFS  *         CALLS  FA=WTW                                                  FA=DEFS      389     A
FA=DEFS                                                                           FA=DEFS      390     A
FA=DEFS                                                                           FA=DEFS      391     A
FA=DEFS            PURGMAC   WRITEW                                               FA=DEFS      392     A
FA=DEFS                                                                           FA=DEFS      393     A
FA=DEFS   WRITEW   MACRO  F,S,N                                                   FA=DEFS      394     A
FA=DEFS            R=     B6,S                                                   FA=DEFS      395     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=DEFS

      FA=DEFS            R=     B7,N                                          FA=DEFS      396     A
      FA=DEFS            R=     A2,F                                          FA=DEFS      397     A
      FA=DEFS            RJ     =XFA=WTW                                      FA=DEFS      398     A
      FA=DEFS   WRITEW   ENDM                                                 FA=DEFS      399     A
      FA=DEFS            SPACE  4                                             FA=DEFS      400     A
      FA=DEFS   #RM      ENDIF                                                FA=DEFS      401     A

           SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - FA=DEFS

                IDENTIFIER                TOTAL    ACTIVE

                FA=DEFS                     401       401
                CCGA018                      31        31

                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=EOF

      FA=EOF    *COMDECK,FA=EOF                                              FA=EOF         1     A
      FA=EOF             CTEXT  FA=EOF - WRITE END-OF-FILE/PARTITION         FA=EOF         2     A
      FA=EOF    EOF      SPACE  4                                             FA=EOF         3     A
      FA=EOF             IF     -DEF,QUAL$,1                                  FA=EOF         4     A
      FA=EOF             QUAL   FA=EOF                                        FA=EOF         5     A
      FA=EOF                                                                  FA=EOF         6     A
      FA=EOF                                                                  FA=EOF         7     A
      FA=EOF    **       FA=EOF - WRITE END-OF-FILE/PARTITION.               FA=EOF         8     A
      FA=EOF    *                                                            FA=EOF         9     A
      FA=EOF    *                                                            FA=EOF        10     A
      FA=EOF    *        ENTRY  (X2) = PSEUDO-FET WORD 1                     FA=EOF        11     A
      FA=EOF    *               (A2) = PSEUDO-FET ADDRESS                    FA=EOF        12     A
      FA=EOF    *               (B1) = 1                                     FA=EOF        13     A
      FA=EOF    *                                                            FA=EOF        14     A
      FA=EOF    *        EXIT   (X2) = PSEUDO-FET WORD 1                     FA=EOF        15     A
      FA=EOF    *               (A2) = PSEUDO-FET ADDRESS                    FA=EOF        16     A
      FA=EOF    *               (B1) = 1                                     FA=EOF        17     A
      FA=EOF    *                                                            FA=EOF        18     A
      FA=EOF    *        USES   X - 7                                        FA=EOF        19     A
      FA=EOF    *               A - NONE                                     FA=EOF        20     A
      FA=EOF    *               B - NONE                                     FA=EOF        21     A
      FA=EOF    *                                                            FA=EOF        22     A
      FA=EOF    *        CALLS  ENDFILE, FA=FLSH                             FA=EOF        23     A
      FA=EOF                                                                  FA=EOF        24     A
      FA=EOF                                                                  FA=EOF        25     A
      FA=EOF    FA=EOF   JP     *+4S15      ** ENTRY/EXIT **                  FA=EOF        26     A
      FA=EOF             PL     X2,EOF2     IF NO HOLDING BUFFER              FA=EOF        27     A
      FA=EOF                                                                  FA=EOF        28     A
      FA=EOF             IFEQ   TEST,0,2                                      FA=EOF        29     A
      FA=EOF             IX7    X2+X2       LEFT ADJUST SPILL FLAG            FA=EOF        30     A
      FA=EOF             PL     X7,FA=EOF   IF FILE NOT SPILLED TO DISK, EXIT FA=EOF        31     A
      FA=EOF                                                                  FA=EOF        32     A
      FA=EOF             RJ     FA=FLSH                                       FA=EOF        33     A
      FA=EOF    EOF2     ENDFILE   X2                                         FA=EOF        34     A
      FA=EOF             JP     FA=EOF      EXIT                              FA=EOF        35     A
      FA=EOF                                                                  FA=EOF        36     A

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
            LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=EOF

FA=EOF                                                                  FA=EOF        37     A
FA=EOF              IF     -DEF,QUAL$,2                                 FA=EOF        38     A
FA=EOF              QUAL   *                                            FA=EOF        39     A
FA=EOF    FA=EOF    EQU    /FA=EOF/FA=EOF                               FA=EOF        40     A
FA=EOF                                                                  FA=EOF        41     A
FA=EOF              IF     -ABS,FA=EOF,1                                FA=EOF        42     A
FA=EOF              ENTRY  FA=EOF                                       FA=EOF        43     A
FA=EOF                                                                  FA=EOF        44     A
FA=EOF              ENDX                                                FA=EOF        45     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - FA=EOF

            IDENTIFIER                TOTAL    ACTIVE

            FA=EOF                       45        45

            LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=EOR

FA=EOR    *COMDECK,FA=EOR                                               FA=EOR         1     A
FA=EOR              CTEXT  FA=EOR - WRITE END-OF-RECORD/SECTION         FA=EOR         2     A
FA=EOR    EOR       SPACE  4                                            FA=EOR         3     A
FA=EOR              IF     -DEF,QUAL$,1                                 FA=EOR         4     A
FA=EOR              QUAL   FA=EOR                                       FA=EOR         5     A
FA=EOR                                                                  FA=EOR         6     A
FA=EOR                                                                  FA=EOR         7     A
FA=EOR    **        FA=EOR - WRITE END-OF-RECORD/SECTION.               FA=EOR         8     A
FA=EOR    *                                                             FA=EOR         9     A
FA=EOR    *                                                             FA=EOR        10     A
FA=EOR    *         ENTRY  (X2) = PSEUDO-FET WORD 1                     FA=EOR        11     A
FA=EOR    *                (A2) = PSEUDO-FET ADDRESS                    FA=EOR        12     A
FA=EOR    *                (B1) = 1                                     FA=EOR        13     A
FA=EOR    *                                                             FA=EOR        14     A
FA=EOR    *         EXIT   (X2) = PSEUDO-FET WORD 1                     FA=EOR        15     A
FA=EOR    *                (A2) = PSEUDO-FET ADDRESS                    FA=EOR        16     A
FA=EOR    *                (B1) = 1                                     FA=EOR        17     A
FA=EOR    *                                                             FA=EOR        18     A
FA=EOR    *         USES   X - 7                                        FA=EOR        19     A
FA=EOR    *                A - NONE                                     FA=EOR        20     A
FA=EOR    *                B - NONE                                     FA=EOR        21     A
FA=EOR    *                                                             FA=EOR        22     A
FA=EOR    *         CALLS  FA=FLSH, WEOR                                FA=EOR        23     A
FA=EOR                                                                  FA=EOR        24     A
FA=EOR                                                                  FA=EOR        25     A
FA=EOR    FA=EOR    JP     *+4S15      ** ENTRY/EXIT **                 FA=EOR        26     A
FA=EOR              PL     X2,EOR2     IF NO HOLDING BUFFER             FA=EOR        27     A
FA=EOR                                                                  FA=EOR        28     A
FA=EOR              IFEQ   TEST,0,2                                     FA=EOR        29     A
FA=EOR              IX7    X2+X2       LEFT ADJUST SPILL FLAG           FA=EOR        30     A
FA=EOR              PL     X7,FA=EOR   IF FILE NOT SPILLED TO DISK, EXIT FA=EOR       31     A
FA=EOR                                                                  FA=EOR        32     A
FA=EOR              RJ     FA=FLSH                                      FA=EOR        33     A
FA=EOR    EOR2      WEOR   X2                                           FA=EOR        34     A

            0         1         2         3         4         5         6         7         8
            12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=EOR

FA=EOR              JP     FA=EOR      EXIT                                    FA=EOR        35     A
FA=EOR                                                                         FA=EOR        36     A
FA=EOR                                                                         FA=EOR        37     A
FA=EOR              IF     -DEF,QUAL$,2                                        FA=EOR        38     A
FA=EOR              QUAL   *                                                   FA=EOR        39     A
FA=EOR    FA=EOR    EQU    /FA=EOR/FA=EOR                                      FA=EOR        40     A
FA=EOR                                                                         FA=EOR        41     A
FA=EOR              IF     -ABS,FA=EOR,1                                       FA=EOR        42     A
FA=EOR              ENTRY  FA=EOR                                              FA=EOR        43     A
FA=EOR                                                                         FA=EOR        44     A
FA=EOR              ENDX                                                       FA=EOR        45     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - FA=EOR

          IDENTIFIER                TOTAL    ACTIVE

          FA=EOR                       45        45

          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=FLSH

FA=FLSH  *COMDECK,FA=FLSH                                                      FA=FLSH        1     A
FA=FLSH             CTEXT  FA=FLSH - FLUSH FILE HOLDING BUFFER                 FA=FLSH        2     A
FA=FLSH   FLSH      SPACE  4                                                   FA=FLSH        3     A
FA=FLSH             IF     -DEF,QUAL$,1                                        FA=FLSH        4     A
FA=FLSH             QUAL   FA=FLSH                                             FA=FLSH        5     A
FA=FLSH                                                                        FA=FLSH        6     A
FA=FLSH                                                                        FA=FLSH        7     A
FA=FLSH  **         FA=FLSH - FLUSH FILE HOLDING BUFFER.                       FA=FLSH        8     A
FA=FLSH  *                                                                     FA=FLSH        9     A
FA=FLSH  *          SEQUENTIAL BINARY FILE -                                   FA=FLSH       10     A
FA=FLSH  *            CALLS RECORD MANAGER TO EMPTY THE HOLDING BUFFER.        FA=FLSH       11     A
FA=FLSH  *            SETS (IN) = (OUT) = (FIRST) IN PSEUDO-FET.               FA=FLSH       12     A
FA=FLSH  *            TURNS SPILL FLAG ON IN PSEUDO-FET.                       FA=FLSH       13     A
FA=FLSH  *                                                                     FA=FLSH       14     A
FA=FLSH  *          OTHER FILE TYPE -                                          FA=FLSH       15     A
FA=FLSH  *            EXITS WITHOUT ACTION (HOLDING BUFFER DOES NOT EXIST).    FA=FLSH       16     A
FA=FLSH  *                                                                     FA=FLSH       17     A
FA=FLSH  *                                                                     FA=FLSH       18     A
FA=FLSH  *          ENTRY  (X2) = PSEUDO-FET WORD 1                            FA=FLSH       19     A
FA=FLSH  *                 (A2) = PSEUDO-FET ADDRESS                           FA=FLSH       20     A
FA=FLSH  *                 (B1) = 1                                            FA=FLSH       21     A
FA=FLSH  *                                                                     FA=FLSH       22     A
FA=FLSH  *          EXIT   (X2) = PSEUDO-FET WORD 1                            FA=FLSH       23     A
FA=FLSH  *                 (A2) = PSEUDO-FET ADDRESS                           FA=FLSH       24     A
FA=FLSH  *                 (B1) = 1                                            FA=FLSH       25     A
FA=FLSH  *                                                                     FA=FLSH       26     A
FA=FLSH  *          USES   X - 1, 3, 4, 6, 7                                   FA=FLSH       27     A
FA=FLSH  *                 A - 1, 3, 4, 6, 7                                   FA=FLSH       28     A
FA=FLSH  *                 B - 7                                               FA=FLSH       29     A
FA=FLSH  *                                                                     FA=FLSH       30     A
FA=FLSH  *          CALLS  PUTW                                                FA=FLSH       31     A
FA=FLSH                                                                        FA=FLSH       32     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=FLSH

```
FA=FLSH                                                                                  FA=FLSH      33     A
FA=FLSH    FA=FLSH  JP     *+4S15      ** ENTRY/EXIT **                                  FA=FLSH      34     A
FA=FLSH             PL     X2,FA=FLSH  IF NO HOLDING BUFFER, EXIT                        FA=FLSH      35     A
FA=FLSH             SA1    A2+B1       (X1) = HOLDING BUFFER -FIRST-                     FA=FLSH      36     A
FA=FLSH             MX7    1                                                             FA=FLSH      37     A
FA=FLSH             SA3    A1+B1       (X3) = HOLDING BUFFER -IN-                        FA=FLSH      38     A
FA=FLSH             LX7    -59D+58D                                                      FA=FLSH      39     A
FA=FLSH             SX6    X1                                                            FA=FLSH      40     A
FA=FLSH             BX7    X7+X2                                                         FA=FLSH      41     A
FA=FLSH             SA6    A3          RESET -IN- = -FIRST-                              FA=FLSH      42     A
FA=FLSH             IX3    X3-X1       NR WORDS IN BUFFER                                FA=FLSH      43     A
FA=FLSH             SA7    A2          TURN SPILL FLAG ON                                FA=FLSH      44     A
FA=FLSH             SA6    A6+B1       RESET -OUT- = -FIRST-                             FA=FLSH      45     A
FA=FLSH             ZR     X3,FA=FLSH  IF HOLDING BUFFER EMPTY, EXIT                     FA=FLSH      46     A
FA=FLSH             FETCH  X2,RT,X4,X4                                                   FA=FLSH      47     A
FA=FLSH             SB7    X4-#ST#                                                       FA=FLSH      48     A
FA=FLSH             ZR     B7,FLSH1    IF S TYPE RECORD                                  FA=FLSH      49     A
FA=FLSH             PUTW   X2,X1,X3                                                      FA=FLSH      50     A
FA=FLSH             JP     FA=FLSH     EXIT                                              FA=FLSH      51     A
FA=FLSH                                                                                  FA=FLSH      52     A
FA=FLSH    FLSH1    PUTWP  X2,X1,X3                                                      FA=FLSH      53     A
FA=FLSH             JP     FA=FLSH     EXIT                                              FA=FLSH      54     A
FA=FLSH                                                                                  FA=FLSH      55     A
FA=FLSH                                                                                  FA=FLSH      56     A
FA=FLSH             IF     -DEF,QUAL$,2                                                  FA=FLSH      57     A
FA=FLSH             QUAL   *                                                             FA=FLSH      58     A
FA=FLSH    FA=FLSH  EQU    /FA=FLSH/FA=FLSH                                              FA=FLSH      59     A
FA=FLSH             ENDX                                                                 FA=FLSH      60     A
```

SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - FA=FLSH

IDENTIFIER	TOTAL	ACTIVE
FA=FLSH	60	60

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=OPE

```
FA=OPE    *COMDECK,FA=OPE                                                                FA=OPE        1     A
FA=OPE              CTEXT  FA=OPE - OPEN FILE                                            FA=OPE        2     A
FA=OPE     OPE      SPACE  4                                                             FA=OPE        3     A
FA=OPE              IF     -DEF,QUAL$,1                                                  FA=OPE        4     A
FA=OPE              QUAL   FA=OPE                                                        FA=OPE        5     A
FA=OPE                                                                                   FA=OPE        6     A
FA=OPE                                                                                   FA=OPE        7     A
FA=OPE    **        FA=OPE - OPEN FILE.                                                  FA=OPE        8     A
FA=OPE    *                                                                              FA=OPE        9     A
FA=OPE    *                OPENS A FILE, RELIABLY.  PREVENTS REDUNDANT OPEN             FA=OPE       10     A
FA=OPE    *         ERRORS.  HONORS AND PRESERVES *PD*, *OF* AND *CF* IN FIT.            FA=OPE       11     A
FA=OPE    *                                                                              FA=OPE       12     A
FA=OPE    *                                                                              FA=OPE       13     A
FA=OPE    *         ENTRY  (X2) = FIT ADDRESS                                            FA=OPE       14     A
FA=OPE    *                                                                              FA=OPE       15     A
```

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=OPE

```
FA=OPE   *         EXIT   (X2) = FIT ADDRESS                                   FA=OPE        16     A
FA=OPE   *                                                                     FA=OPE        17     A
FA=OPE   *         USES   X - 3, 6                                             FA=OPE        18     A
FA=OPE   *                A - 3                                                FA=OPE        19     A
FA=OPE   *                B - 2                                                FA=OPE        20     A
FA=OPE   *                                                                     FA=OPE        21     A
FA=OPE   *         SAVES  (A5), (X5)                                           FA=OPE        22     A
FA=OPE   *                                                                     FA=OPE        23     A
FA=OPE   *         CALLS  CLOSEM, FETCH, OPENM, SETFIT, STORE                  FA=OPE        24     A
FA=OPE                                                                         FA=OPE        25     A
FA=OPE                                                                         FA=OPE        26     A
FA=OPE    FA=OPE   JP     *+4S15      ** ENTRY/EXIT **                         FA=OPE        27     A
FA=OPE                                                                         FA=OPE        28     A
FA=OPE   *         CHECK FILE OPEN/CLOSED STATUS.                              FA=OPE        29     A
FA=OPE                                                                         FA=OPE        30     A
FA=OPE             FETCH  X2,PD,X6,X3                                          FA=OPE        31     A
FA=OPE             SB2    X6          TEMP SAVE *PD* (*SETFIT* WILL DESTROY)   FA=OPE        32     A
FA=OPE             SETFIT X2          FDT PARAMETERS TO FIT                    FA=OPE        33     A
FA=OPE             FETCH  X2,OC,X6,X3                                          FA=OPE        34     A
FA=OPE             SX6    X6-#OPE#                                            FA=OPE        35     A
FA=OPE             NZ     X6,OPE2     IF FILE NOT OPEN                        FA=OPE        36     A
FA=OPE                                                                         FA=OPE        37     A
FA=OPE   *         HERE IF FILE LEFT OPEN BY PREVIOUS JOB STEP.  SOME ATTRIBUTES FA=OPE      38     A
FA=OPE   *         OF AN OPEN FILE, SUCH AS *PD*, CANNOT BE CHANGED, SO WE CLOSE FA=OPE      39     A
FA=OPE   *         IT FIRST (SIGH).                                            FA=OPE        40     A
FA=OPE                                                                         FA=OPE        41     A
FA=OPE             FETCH  X2,CF,X6,X3 TEMP SAVE *CF* (*CLOSEM* WILL DESTROY)   FA=OPE        42     A
FA=OPE             CLOSEM X2,N                                                 FA=OPE        43     A
FA=OPE             STORE  X2,CF=X6,X3                                          FA=OPE        44     A
FA=OPE                                                                         FA=OPE        45     A
FA=OPE   *         (RE)OPEN FILE.                                              FA=OPE        46     A
FA=OPE                                                                         FA=OPE        47     A
FA=OPE    OPE2     SX6    B2                                                  FA=OPE        48     A
FA=OPE             STORE  X2,PD=X6,X3                                          FA=OPE        49     A
FA=OPE             OPENM  X2                                                   FA=OPE        50          I
          -CCGA018
FA=OPE                                                                         CCGA018       72     A
FA=OPE   **        ** NASTY (TEMPORARY) KLUDGE ** FOLLOWS                      CCGA018       73     A
FA=OPE                                                                         CCGA018       74     A
FA=OPE             SA3    =XCO.REWI                                           CCGA018       75     A
FA=OPE             PL     X3,OPE5     IF REW OPTION NOT ON                    CCGA018       76     A
FA=OPE             SA3    =XF.IN                                              CCGA018       77     A
FA=OPE             BX3    X3-X2                                               CCGA018       78     A
FA=OPE             SX3    X3                                                  CCGA018       79     A
FA=OPE             NZ     X3,OPE5     IF NOT FILE INPUT                       CCGA018       80     A
FA=OPE             STORE  X2,OF=R                                             CCGA018       81     A
FA=OPE                                                                         CCGA079        6     A
FA=OPE    OPE5     SA3    =XCO.REWB                                           CCGA079        7     A
FA=OPE             PL     X3,OPE6     IF REW OPTION NOT ON                    CCGA079        8     A
FA=OPE             SA3    =XF.LGO                                             CCGA079        9     A
FA=OPE             BX3    X3-X2                                               CCGA079       10     A
FA=OPE             SX3    X3                                                  CCGA079       11     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=OPE

FA=OPE              NZ     X3,OPE6     IF NOT FILE LGO                          CCGA079      12     A
FA=OPE              STORE  X2,OF=R                                              CCGA079      13     A
FA=OPE                                                                          CCGA018      82     A
FA=OPE    **        END OF NASTY KLUDGE                                         CCGA018      83     A
FA=OPE                                                                          CCGA018      84     A
FA=OPE     OPE5     OPENM  X2                                                   CCGA018      85          I
          -CCGA079
FA=OPE     OPE6     OPENM  X2                                                   CCGA079      14     A
FA=OPE              JP     FA=OPE      EXIT                                     FA=OPE       51     A
FA=OPE                                                                          FA=OPE       52     A
FA=OPE                                                                          FA=OPE       53     A
FA=OPE              IF     -DEF,QUAL$,2                                         FA=OPE       54     A
FA=OPE              QUAL   *                                                    FA=OPE       55     A
FA=OPE     FA=OPE   EQU    /FA=OPE/FA=OPE                                       FA=OPE       56     A
FA=OPE                                                                          FA=OPE       57     A
FA=OPE              IF     -ABS,FA=OPE,1                                        FA=OPE       58     A
FA=OPE              ENTRY  FA=OPE                                               FA=OPE       59     A
FA=OPE                                                                          FA=OPE       60     A
FA=OPE              ENDX                                                        FA=OPE       61     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - FA=OPE

          IDENTIFIER                TOTAL    ACTIVE

          FA=OPE                       61        60
          CCGA018                      14        13
          CCGA079                       9         9

          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=RDC

FA=RDC    *COMDECK,FA=RDC                                                       FA=RDC        1     A
FA=RDC              CTEXT  FA=RDC - READ CODED LINE, -C- FORMAT                 FA=RDC        2     A
FA=RDC     RDC      SPACE  4                                                    FA=RDC        3     A
FA=RDC              IF     -DEF,QUAL$,1                                         FA=RDC        4     A
FA=RDC              QUAL   FA=RDC                                               FA=RDC        5     A
FA=RDC                                                                          FA=RDC        6     A
FA=RDC                                                                          FA=RDC        7     A
FA=RDC    **        FA=RDC - READ ONE CODED LINE FROM FILE, IN -C- FORMAT.      FA=RDC        8     A
FA=RDC    *                                                                     FA=RDC        9     A
FA=RDC    *                                                                     FA=RDC       10     A
FA=RDC    *         ENTRY  (X2) = PSEUDO-FET WORD 1.                            FA=RDC       11     A
FA=RDC    *                (A2) = PSEUDO-FET ADDRESS.                           FA=RDC       12     A
FA=RDC    *                (B1) = 1                                             FA=RDC       13     A
FA=RDC    *                (B6) = WSA ADDRESS.                                  FA=RDC       14     A
FA=RDC    *                (B7) = WSA LENGTH (WORDS).                           FA=RDC       15     A
FA=RDC    *                                                                     FA=RDC       16     A
FA=RDC    *         EXIT   (X1) = 0 FOR TRANSFER COMPLETE.                      FA=RDC       17     A
FA=RDC    *                (X1) = -1 IF EOP/EOI DETECTED ON FILE.               FA=RDC       18     A
FA=RDC    *                (X1) = (B6) IF EOS DETECTED ON FILE BEFORE TRANSFER  FA=RDC       19     A
FA=RDC    *                           WAS COMPLETED.                            FA=RDC       20     A
FA=RDC    *                (X2) = ADDRESS OF FIT FOR FILE.                      FA=RDC       21     A
FA=RDC    *                (B1) = 1                                             FA=RDC       22     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=RDC

```
FA=RDC   *                (B6) = ADDRESS PLUS ONE OF LAST WORD TRANSFERRED TO          FA=RDC     23    A
FA=RDC   *                          WORKING STORAGE AREA.                              FA=RDC     24    A
FA=RDC   *                                                                             FA=RDC     25    A
FA=RDC   *         USES   X - 1, 3, 4, 6, 7                                            FA=RDC     26    A
FA=RDC   *                A - 1, 3, 6, 7                                               FA=RDC     27    A
FA=RDC   *                B - 2, 6                                                     FA=RDC     28    A
FA=RDC   *                                                                             FA=RDC     29    A
FA=RDC   *         CALLS  FETCH, GETW, STORE                                           FA=RDC     30    A
FA=RDC                                                                                 CCGA018    86    A
FA=RDC    RDCA     BSS    1           PRESERVE X5 HERE                                 CCGA018    87    A
FA=RDC    RDCB     BSS    1           PRESERVE A5 HERE                                 CCGA018    88    A
FA=RDC                                                                                 FA=RDC     31    A
FA=RDC                                                                                 FA=RDC     32    A
FA=RDC    FA=RDC   JP     *+4S15      ** ENTRY/EXIT **                                 FA=RDC     33    A
FA=RDC                                                                                 FA=RDC     34    A
FA=RDC             IFNE   TEST,0,1                                                     FA=RDC     35    A
FA=RDC             LE     B7,B0,*+4S15      IF LENGTH ERROR, ABORT JOB                  FA=RDC     36    A
FA=RDC                                                                                 FA=RDC     37    A
FA=RDC             SX3    B6          (X3) = WSA ADDRESS                               FA=RDC     38    A
FA=RDC             SX4    B7          (X4) = WSA LENGTH                                FA=RDC     39    A
FA=RDC             GETW   X2,X3,X4,RDC2                                                FA=RDC     40    A
FA=RDC             SB6    B6+X4       (B6) = LWA+1 OF TRANSFER                         FA=RDC     41    A
FA=RDC             AX4    18D         (X4) = UNUSED BIT COUNT                          FA=RDC     42    A
FA=RDC             SA3    B6-B1       LAST WORD OF LINE                                FA=RDC     43    A
FA=RDC             MX6    1                                                            FA=RDC     44    A
FA=RDC             SB2    X4-59D                                                       FA=RDC     45    A
FA=RDC             MX7    0                                                            FA=RDC     46    A
FA=RDC             AX4    X6,-B2                                                       FA=RDC     47    A
FA=RDC             MX1    0           *TRANSFER COMPLETE* STATUS                       FA=RDC     48    A
FA=RDC             BX6    X4*X3       DISCARD UNUSED BITS                              FA=RDC     49    A
FA=RDC             AX4    6                                                            FA=RDC     50    A
FA=RDC             SA6    A3          TERMINATE LAST WORD READ                         FA=RDC     51    A
FA=RDC             NZ     X4,FA=RDC   IF AT LEAST 12-BIT LINE TERMINATOR, EXIT         FA=RDC     52    A
FA=RDC             SB6    B6+B1       INCREMENT LWA+1 OF TRANSFER                      FA=RDC     53    A
FA=RDC             SA7    A6+B1       TERMINATE LINE                                   FA=RDC     54    A
FA=RDC             JP     FA=RDC      EXIT                                             FA=RDC     55    A
FA=RDC                                                                                 FA=RDC     56    A
FA=RDC   *         HERE IF END-OF-DATA ENCOUNTERED.                                    FA=RDC     57    A
FA=RDC                                                                                 FA=RDC     58    A
FA=RDC    RDC2     JP     *+4S15      ** REC MGR SIMULATES -RJ- HERE IF E-O-D **       FA=RDC     59    A
FA=RDC             FETCH  X2,FP,X3,X1                                                  FA=RDC     60          I
          -CCGA018
FA=RDC             SX3    X3-#EOS#                                                     FA=RDC     61          I
          -CCGA018
FA=RDC             SB6    B6+X4       (B6) = LWA+1 OF TRANSFER                         FA=RDC     62          I
          -CCGA018
FA=RDC             SX1    B6          TO RETURN *EOS* STATUS                           FA=RDC     63          I
          -CCGA018
FA=RDC             ZR     X3,FA=RDC   IF EOS, EXIT                                     FA=RDC     64          I
          -CCGA018
FA=RDC             BX6    X5                                                           CCGA018    89    A
FA=RDC             SA6    RDCA        PRESERVE X5                                      CCGA018    90    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=RDC

FA=RDC            SX6    A5                                                CCGA018      91    A
FA=RDC            SA6    RDCB        PRESERVE A5                           CCGA018      92    A
FA=RDC            FETCH  X2,FP,X7,X1                                       CCGA018      93    A
FA=RDC            SA3    RDCB                                              CCGA018      94    A
FA=RDC            SA5    X3          RESTORE A5                            CCGA018      95    A
FA=RDC            SA3    RDCA                                              CCGA018      96    A
FA=RDC            BX5    X3          RESTORE X5                            CCGA018      97    A
FA=RDC            SX7    X7-#EOS#                                          CCGA018      98    A
FA=RDC            SB6    B6+X4       (B6) = LWA+1 OF TRANSFER              CCGA018      99    A
FA=RDC            SX1    B6          TO RETURN *EOS* STATUS                CCGA018     100    A
FA=RDC            ZR     X7,FA=RDC   IF EOS, EXIT                          CCGA018     101    A
FA=RDC            MX1    -1          RETURN EOP/EOI STATUS                 FA=RDC       65    A
FA=RDC            JP     FA=RDC      EXIT                                  FA=RDC       66    A
FA=RDC                                                                     FA=RDC       67    A
FA=RDC                                                                     FA=RDC       68    A
FA=RDC            IF     -DEF,QUAL$,2                                      FA=RDC       69    A
FA=RDC            QUAL   *                                                 FA=RDC       70    A
FA=RDC    FA=RDC  EQU    /FA=RDC/FA=RDC                                    FA=RDC       71    A
FA=RDC                                                                     FA=RDC       72    A
FA=RDC            IF     -ABS,FA=RDC,1                                     FA=RDC       73    A
FA=RDC            ENTRY  FA=RDC                                            FA=RDC       74    A
FA=RDC                                                                     FA=RDC       75    A
FA=RDC            ENDX                                                     FA=RDC       76    A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - FA=RDC

          IDENTIFIER                TOTAL    ACTIVE

          FA=RDC                       76        71
          CCGA018                      16        16

          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=RDW

FA=RDW    *COMDECK,FA=RDW                                                  FA=RDW        1    A
FA=RDW            CTEXT  FA=RDW - READ WORDS TO WORKING BUFFER             FA=RDW        2    A
FA=RDW     RDW    SPACE  4                                                 FA=RDW        3    A
FA=RDW            IF     -DEF,QUAL$,1                                      FA=RDW        4    A
FA=RDW            QUAL   FA=RDW                                            FA=RDW        5    A
FA=RDW                                                                     FA=RDW        6    A
FA=RDW                                                                     FA=RDW        7    A
FA=RDW    **      FA=RDW - READ WORDS TO WORKING BUFFER.                   FA=RDW        8    A
FA=RDW    *                                                                FA=RDW        9    A
FA=RDW    *              *FA=RDW* PARALLELS ITS KRONOS MODEL, *COMCRDW* AS FA=RDW       10    A
FA=RDW    *       CLOSELY AS POSSIBLE.  RECORD MANAGER COMPLICATED THINGS BY FA=RDW      11    A
FA=RDW    *       ADDING A NEW LEVEL, *RECORD*, TO THE LOGICAL STRUCTURE   FA=RDW       12    A
FA=RDW    *       HIERARCHY OF FILES.  SINCE CIO-STYLED I/O HAS NO SUCH LEVEL FA=RDW    13    A
FA=RDW    *       (A CIO *RECORD* RESEMBLES A RECORD MANAGER *SECTION*),   FA=RDW       14    A
FA=RDW    *       *FA=RDW* IGNORES 7RM RECORD BOUNDARIES AND RETURNS AN    FA=RDW       15    A
FA=RDW    *       END-OF-DATA STATUS ONLY WHEN END-OF-SECTION IS SENSED.   FA=RDW       16    A
FA=RDW    *                                                                FA=RDW       17    A
FA=RDW    *                                                                FA=RDW       18    A
FA=RDW    *       ENTRY  (X2) = PSEUDO-FET WORD 1.                         FA=RDW       19    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=RDW

```
FA=RDW   *                (A2) = PSEUDO-FET ADDRESS.                                FA=RDW      20    A
FA=RDW   *                (B1) = 1                                                  FA=RDW      21    A
FA=RDW   *                (B6) = WSA ADDRESS.                                       FA=RDW      22    A
FA=RDW   *                (B7) = WSA LENGTH (WORDS).                                FA=RDW      23    A
FA=RDW   *                                                                          FA=RDW      24    A
FA=RDW   *         EXIT   (X1) = 0 FOR TRANSFER COMPLETE.                           FA=RDW      25    A
FA=RDW   *                (X1) = -1 IF EOP/EOI DETECTED ON FILE.                    FA=RDW      26    A
FA=RDW   *                (X1) = (B6) IF EOS DETECTED ON FILE BEFORE TRANSFER       FA=RDW      27    A
FA=RDW   *                            WAS COMPLETED.                                FA=RDW      28    A
FA=RDW   *                (X2) = PSEUDO-FET WORD 1.                                 FA=RDW      29    A
FA=RDW   *                (A2) = PSEUDO-FET ADDRESS.                                FA=RDW      30    A
FA=RDW   *                (B1) = 1                                                  FA=RDW      31    A
FA=RDW   *                (B6) = ADDRESS PLUS ONE OF LAST WORD TRANSFERRED TO       FA=RDW      32    A
FA=RDW   *                            WORKING STORAGE AREA.                         FA=RDW      33    A
FA=RDW   *                                                                          FA=RDW      34    A
FA=RDW   *         USES   X - 1, 2, 3, 4, 6, 7                                      FA=RDW      35    A
FA=RDW   *                A - 2, 3, 4, 6, 7                                         FA=RDW      36    A
FA=RDW   *                B - 2, 3, 4, 6, 7                                         FA=RDW      37    A
FA=RDW   *                                                                          FA=RDW      38    A
FA=RDW   *         CALLS  FETCH, GETWP, MOVE                                        FA=RDW      39    A
FA=RDW                                                                              CCGA018    102    A
FA=RDW    RDWA     BSS    1           PRESERVE X5 HERE                              CCGA018    103    A
FA=RDW    RDWB     BSS    1           PRESERVE A5 HERE                              CCGA018    104    A
FA=RDW                                                                              FA=RDW      40    A
FA=RDW                                                                              FA=RDW      41    A
FA=RDW    FA=RDW   JP     *+4S15      ** ENTRY/EXIT **                              FA=RDW      42    A
FA=RDW                                                                              FA=RDW      43    A
FA=RDW             IFNE   TEST,0,1                                                  FA=RDW      44    A
FA=RDW             LE     B7,B0,*+4S15      IF LENGTH ERROR, ABORT JOB               FA=RDW      45    A
FA=RDW                                                                              FA=RDW      46    A
FA=RDW             SB2    A2          (B2) = PSEUDO-FET ADDRESS                     FA=RDW      47    A
FA=RDW             SB3    B7+         (B3) = NR UNTRANSFERRED WORDS                 FA=RDW      48    A
FA=RDW             SB4    B6          (B4) = FWA OF TRANSFER                        FA=RDW      49    A
FA=RDW    RDW2     SA3    B2+I.IN     (X3) = HOLDING BUFFER -IN- POINTER            FA=RDW      50    A
FA=RDW             SA2    A3+B1       (X2) = HOLDING BUFFER -OUT- POINTER           FA=RDW      51    A
FA=RDW             SX6    B3          NR WORDS REQUESTED                            FA=RDW      52    A
FA=RDW             IX7    X3-X2       NR WORDS IN HOLDING BUFFER                    FA=RDW      53    A
FA=RDW             ZR     X7,RDW6     IF HOLDING BUFFER EMPTY                       FA=RDW      54    A
FA=RDW                                                                              FA=RDW      55    A
FA=RDW   *         COMPUTE MINIMUM OF NUMBER OF WORDS REQUESTED VS. WORDS IN        FA=RDW      56    A
FA=RDW   *         HOLDING BUFFER.  TRANSFER *MIN* WORDS TO WORKING STORAGE AREA.   FA=RDW      57    A
FA=RDW                                                                              FA=RDW      58    A
FA=RDW             IX1    X6-X7                                                     FA=RDW      59    A
FA=RDW             SX3    B6          (X3) = FWA OF TRANSFER                        FA=RDW      60    A
FA=RDW             AX1    59D                                                       FA=RDW      61    A
FA=RDW             BX6    X1*X6                                                     FA=RDW      62    A
FA=RDW             BX7    -X1*X7                                                    FA=RDW      63    A
FA=RDW             IX1    X6+X7       (X1) = MIN (REQUESTED, AVAILABLE) NR WORDS    FA=RDW      64    A
FA=RDW             SB7    X1                                                        FA=RDW      65    A
FA=RDW             IX6    X2+X1                                                     FA=RDW      66    A
FA=RDW             SB3    B3-B7                                                     FA=RDW      67    A
FA=RDW             SB6    B6+X1       (B6) = LWA+1 OF TRANSFER                      FA=RDW      68    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=RDW

```
FA=RDW            SA6    A2          UPDATE -OUT-                                  FA=RDW      69    A
FA=RDW            MOVE   X1,X2,X3                                                  FA=RDW      70    A
FA=RDW            SA2    B2          RESTORE (X2) = FILE CONTROL WORD              FA=RDW      71    A
FA=RDW            MX1    0           *TRANSFER COMPLETE* STATUS                    FA=RDW      72    A
FA=RDW            ZR     B3,FA=RDW   IF TRANSFER COMPLETE, EXIT                    FA=RDW      73    A
FA=RDW            LX3    X2,B1       LEFT ADJUST SPILL FLAG                        FA=RDW      74    A
FA=RDW            SX1    B6          *EOS DETECTED* STATUS                         FA=RDW      75    A
FA=RDW            PL     X3,FA=RDW   IF FILE NOT SPILLED TO DISK, EXIT             FA=RDW      76    A
FA=RDW            FETCH  X2,FP,X3,X4                                               FA=RDW      77          I
          -CCGA018
FA=RDW            BX6    X5                                                        CCGA018    105    A
FA=RDW            SA6    RDWA        PRESERVE X5                                   CCGA018    106    A
FA=RDW            SX6    A5                                                        CCGA018    107    A
FA=RDW            SA6    RDWB        PRESERVE A5                                   CCGA018    108    A
FA=RDW            FETCH  X2,FP,X3,X4                                               CCGA018    109    A
FA=RDW            SA4    RDWB                                                      CCGA018    110    A
FA=RDW            SA5    X4          RESTORE A5                                    CCGA018    111    A
FA=RDW            SA4    RDWA                                                      CCGA018    112    A
FA=RDW            BX5    X4          RESTORE X5                                    CCGA018    113    A
FA=RDW            SX4    #EOS#+#EOP#+#EOI#                                         FA=RDW      78    A
FA=RDW            BX3    X4*X3                                                     FA=RDW      79    A
FA=RDW            NZ     X3,RDW5     IF FILE AT EOS/EOP/EOI                        FA=RDW      80    A
FA=RDW                                                                             FA=RDW      81    A
FA=RDW    *       REFILL HOLDING BUFFER.                                           FA=RDW      82    A
FA=RDW                                                                             FA=RDW      83    A
FA=RDW     RDW3   SA3    B2+B1       -FIRST-                                       FA=RDW      84    A
FA=RDW            SA4    B2+I.LIMIT  -LIMIT-                                       FA=RDW      85    A
FA=RDW            IX1    X4-X3       HOLDING BUFFER LENGTH                         FA=RDW      86    A
FA=RDW            GETWP  X2,X3,X1,RDW7     RETURNS (X1) = TRANSFER LENGTH          FA=RDW      87    A
FA=RDW     RDW4   ZR     X1,RDW5     IF NO DATA RECEIVED                           FA=RDW      88    A
FA=RDW            BX6    X3                                                        FA=RDW      89    A
FA=RDW            IX7    X3+X1                                                     FA=RDW      90    A
FA=RDW            SA6    A4-B1       RESET -OUT- = -FIRST-                         FA=RDW      91    A
FA=RDW            SA7    A3+B1       SET NEW -IN-                                  FA=RDW      92    A
FA=RDW            JP     RDW2        CONTINUE TRANSFER                             FA=RDW      93    A
FA=RDW                                                                             FA=RDW      94    A
FA=RDW    *       PROCESS END-OF-DATA CONDITION.                                   FA=RDW      95    A
FA=RDW                                                                             FA=RDW      96    A
FA=RDW     RDW5   SA3    B2+B1       -FIRST-                                       FA=RDW      97    A
FA=RDW            SX1    B6          *EOS DETECTED* STATUS                         FA=RDW      98    A
FA=RDW            BX6    X3                                                        FA=RDW      99    A
FA=RDW            SA6    A3+B1       RESET -IN- = -FIRST-                          FA=RDW     100    A
FA=RDW            SA6    A6+B1       RESET -OUT- = -FIRST-                         FA=RDW     101    A
FA=RDW            GT     B6,B4,FA=RDW     IF ANY DATA TRANSFERRED TO WSA           FA=RDW     102    A
FA=RDW            MX1    -1          *EOP/EOI* STATUS                              FA=RDW     103    A
FA=RDW            JP     FA=RDW      EXIT                                          FA=RDW     104    A
FA=RDW                                                                             FA=RDW     105    A
FA=RDW    *       PROCESS EMPTY HOLDING BUFFER.                                    FA=RDW     106    A
FA=RDW                                                                             FA=RDW     107    A
FA=RDW     RDW6   SA2    B2          RESTORE (X2) = PSEUDO-FET WORD 1              FA=RDW     108    A
FA=RDW            LX1    X2,B1       LEFT ADJUST SPILL FLAG                        FA=RDW     109    A
FA=RDW            PL     X1,RDW5     IF FILE NOT SPILLED TO DISK                   FA=RDW     110    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=RDW

FA=RDW              JP     RDW3        GO REFILL HOLDING BUFFER                          FA=RDW       111     A
FA=RDW                                                                                   FA=RDW       112     A
FA=RDW    *         PROCESS RECORD MANAGER DATA EXIT.                                    FA=RDW       113     A
FA=RDW                                                                                   FA=RDW       114     A
FA=RDW     RDW7     JP     *+4S15      ** REC MGR SIMULATES -RJ- HERE IF E-O-D **        FA=RDW       115     A
FA=RDW              JP     RDW4        CONTINUE (IGNORE 7RM *RECORD* END)                FA=RDW       116     A
FA=RDW                                                                                   FA=RDW       117     A
FA=RDW                                                                                   FA=RDW       118     A
FA=RDW              IF     -DEF,QUAL$,2                                                  FA=RDW       119     A
FA=RDW              QUAL   *                                                             FA=RDW       120     A
FA=RDW     FA=RDW   EQU    /FA=RDW/FA=RDW                                                FA=RDW       121     A
FA=RDW                                                                                   FA=RDW       122     A
FA=RDW              IF     -ABS,FA=RDW,1                                                 FA=RDW       123     A
FA=RDW              ENTRY  FA=RDW                                                        FA=RDW       124     A
FA=RDW                                                                                   FA=RDW       125     A
FA=RDW              ENDX                                                                 FA=RDW       126     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - FA=RDW

          IDENTIFIER                TOTAL    ACTIVE

          FA=RDW                      126       125
          CCGA018                      12        12

          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=RWX

FA=RWX    *COMDECK,FA=RWX                                                                FA=RWX         1     A
FA=RWX              CTEXT  FA=RWX - REWIND FILE (FANCY)                                  FA=RWX         2     A
FA=RWX     RWX      SPACE  4                                                             FA=RWX         3     A
FA=RWX              IF     -DEF,QUAL$,1                                                  FA=RWX         4     A
FA=RWX              QUAL   FA=RWX                                                        FA=RWX         5     A
FA=RWX                                                                                   FA=RWX         6     A
FA=RWX                                                                                   FA=RWX         7     A
FA=RWX    **        FA=RWX - REWIND FILE (SUPPRESSED FOR SCRATCH FILES).                 FA=RWX         8     A
FA=RWX    *                                                                              FA=RWX         9     A
FA=RWX    *         SEQUENTIAL BINARY FILE -                                             FA=RWX        10     A
FA=RWX    *           REWINDS THE FILE, UNLESS THE HOLDING BUFFER HAS NOT                FA=RWX        11     A
FA=RWX    *           OVERFLOWED AND THE COMPILER IS ASSEMBLED IN SYSTEM MODE.           FA=RWX        12     A
FA=RWX    *                                                                              FA=RWX        13     A
FA=RWX    *         OTHER FILE TYPE -                                                    FA=RWX        14     A
FA=RWX    *           REWINDS THE FILE UNCONDITIONALLY.                                  FA=RWX        15     A
FA=RWX    *                                                                              FA=RWX        16     A
FA=RWX    *                                                                              FA=RWX        17     A
FA=RWX    *         ENTRY  (X2) = PSEUDO-FET WORD 1                                      FA=RWX        18     A
FA=RWX    *                (A2) = PSEUDO-FET ADDRESS                                     FA=RWX        19     A
FA=RWX    *                                                                              FA=RWX        20     A
FA=RWX    *         EXIT   (X2) = PSEUDO-FET WORD 1                                      FA=RWX        21     A
FA=RWX    *                (A2) = PSEUDO-FET ADDRESS                                     FA=RWX        22     A
FA=RWX    *                                                                              FA=RWX        23     A
FA=RWX    *         USES   X - 1, 6, 7                                                   FA=RWX        24     A
FA=RWX    *                A - 1, 6                                                      FA=RWX        25     A
FA=RWX    *                B - NONE                                                      FA=RWX        26     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=RWX

      FA=RWX    *                                                          FA=RWX        27     A
      FA=RWX    *         CALLS   REWINDM                                  FA=RWX        28     A
      FA=RWX                                                               FA=RWX        29     A
      FA=RWX                                                               FA=RWX        30     A
      FA=RWX     FA=RWX   JP      *+4S15      ** ENTRY/EXIT **            FA=RWX        31     A
      FA=RWX              PL      X2,RWX2     IF FILE HAS NO HOLDING BUFFER FA=RWX      32     A
      FA=RWX              SA1     A2+I.FIRST                               FA=RWX        33     A
      FA=RWX              IX7     X2+X2       LEFT ADJUST SPILL FLAG       FA=RWX        34     A
      FA=RWX              BX6     X1                                       FA=RWX        35     A
      FA=RWX              SA6     A2+I.OUT    SET -OUT- = -FIRST-          FA=RWX        36     A
      FA=RWX                                                               FA=RWX        37     A
      FA=RWX              IFEQ    TEST,0,1                                 FA=RWX        38     A
      FA=RWX              PL      X7,FA=RWX   IF HOLDING BUFFER HAS NOT SPILLED, EXIT FA=RWX 39 A
      FA=RWX                                                               FA=RWX        40     A
      FA=RWX              SA6     A2+I.IN     SET -IN- = -FIRST-           FA=RWX        41     A
      FA=RWX     RWX2     REWINDM   X2                                     FA=RWX        42     A
      FA=RWX              JP      FA=RWX      EXIT                         FA=RWX        43     A
      FA=RWX                                                               FA=RWX        44     A
      FA=RWX                                                               FA=RWX        45     A
      FA=RWX              IF      -DEF,QUAL$,2                             FA=RWX        46     A
      FA=RWX              QUAL    *                                        FA=RWX        47     A
      FA=RWX     FA=RWX   EQU     /FA=RWX/FA=RWX                           FA=RWX        48     A
      FA=RWX                                                               FA=RWX        49     A
      FA=RWX              IF      -ABS,FA=RWX,1                            FA=RWX        50     A
      FA=RWX              ENTRY   FA=RWX                                   FA=RWX        51     A
      FA=RWX                                                               FA=RWX        52     A
      FA=RWX              ENDX                                             FA=RWX        53     A

          SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - FA=RWX

                IDENTIFIER                TOTAL    ACTIVE

                FA=RWX                       53        53

                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=SET

      FA=SET    *COMDECK FA=SET                                            FA=SET         1     A
      FA=SET              CTEXT   FA=SET - SET FILE TABLES                 FA=SET         2     A
      FA=SET     SET      SPACE   4                                        FA=SET         3     A
      FA=SET              IF      -DEF,QUAL$,1                             FA=SET         4     A
      FA=SET              QUAL    FA=SET                                   FA=SET         5     A
      FA=SET                                                               FA=SET         6     A
      FA=SET                                                               FA=SET         7     A
      FA=SET     #RM      IFEQ    CP#RM,0                                  FA=SET         8     A
      FA=SET                                                               FA=SET         9     A
      FA=SET    **        FA=SET - SET FILE ENVIRONMENT TABLE  (CIO DIRECT I/O). FA=SET 10 A
      FA=SET    *                                                          FA=SET        11     A
      FA=SET    *         SETS CIO BUFFER ADDRESSES IN FET.  EITHER THE FWA OR LWA+1 OF FA=SET 12 A
      FA=SET    *         THE BUFFER SPACE IS REQUIRED AS AN INPUT PARAMETER.  THE OTHER FA=SET 13 A
      FA=SET    *         ADDRESS IS COMPUTED.  BOTH ADDRESSES ARE IN REGISTERS ON EXIT. FA=SET 14 A
      FA=SET    *         BUFFER LENGTH IS OBTAINED FROM THE FET (I.CBSET).  FA=SET 15 A
      FA=SET    *                                                          FA=SET        16     A

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=SET

```
FA=SET   *                                                                  FA=SET     17   A
FA=SET   *        ENTRY  (X2) = FET ADDRESS                                 FA=SET     18   A
FA=SET   *               (B1) = 1                                           FA=SET     19   A
FA=SET   *               (B2) = BUFFER FWA OR 0                             FA=SET     20   A
FA=SET   *               (B3) = 0 OR BUFFER LWA+1                           FA=SET     21   A
FA=SET   *                                                                  FA=SET     22   A
FA=SET   *        EXIT   (X2) = FET ADDRESS                                 FA=SET     23   A
FA=SET   *               (B1) = 1                                           FA=SET     24   A
FA=SET   *               (B2) = BUFFER FWA                                  FA=SET     25   A
FA=SET   *               (B3) = BUFFER LWA+1                                FA=SET     26   A
FA=SET   *                                                                  FA=SET     27   A
FA=SET   *        USES   X - 1, 3, 4, 6, 7                                  FA=SET     28   A
FA=SET   *               A - 1, 4, 6, 7                                     FA=SET     29   A
FA=SET   *               B - 2, 3                                           FA=SET     30   A
FA=SET   *                                                                  FA=SET     31   A
FA=SET   *        CALLS  NONE                                               FA=SET     32   A
FA=SET                                                                      FA=SET     33   A
FA=SET                                                                      FA=SET     34   A
FA=SET    FA=SET  JP     *+4S15      ** ENTRY/EXIT **                       FA=SET     35   A
FA=SET            SA1    X2          (X1) = FET WORD 1                      FA=SET     36   A
FA=SET            LX1    59-0                                               FA=SET     37   A
FA=SET            MI     X1,SET2     IF FILE QUIET                          FA=SET     38   A
FA=SET            RECALL X2          WAIT FOR FILE QUIET; RETURNS (X2) = FET ADR  FA=SET  39  A
FA=SET                                                                      FA=SET     40   A
FA=SET   *        NEXT SEQUENCE SETS (X6)=FWA AND (X7)=LWA+1, GIVEN EITHER  FA=SET     41   A
FA=SET   *        (B2)=FWA AND (B3)=0,  OR,  (B2)=0 AND (B3)=LWA+1.         FA=SET     42   A
FA=SET                                                                      FA=SET     43   A
FA=SET    SET2    SA1    X2+I.CBSET                                         FA=SET     44   A
FA=SET            SX3    B2-B3                                              FA=SET     45   A
FA=SET            AX3    59D                                                FA=SET     46   A
FA=SET            AX1    18          (X1) = CIO BUFFER LENGTH               FA=SET     47   A
FA=SET            BX6    -X1*X3                                             FA=SET     48   A
FA=SET            NO                                                        FA=SET     49   A
FA=SET            BX7    -X3*X1                                             FA=SET     50   A
FA=SET            SX3    B2+B3       FWA OR LWA+1                           FA=SET     51   A
FA=SET            IX6    X6+X3       (X6) = BUFFER FWA                      FA=SET     52   A
FA=SET            LX1    18                                                 FA=SET     53   A
FA=SET            SB2    X6          (B2) = BUFFER FWA                      FA=SET     54   A
FA=SET            IX7    X7+X3       (X7) = BUFFER LWA+1                    FA=SET     55   A
FA=SET                                                                      FA=SET     56   A
FA=SET   *        STORE VALUES IN FET.                                      FA=SET     57   A
FA=SET                                                                      FA=SET     58   A
FA=SET            SA6    X2+I.OUT    OUT                                    FA=SET     59   A
FA=SET            SB3    X7          (B3) = BUFFER LWA+1                    FA=SET     60   A
FA=SET            SA4    X2+I.LIMIT  FET+4                                  FA=SET     61   A
FA=SET            MX3    42                                                 FA=SET     62   A
FA=SET            BX4    X4*X3       SAVE PRU SIZE ETC.                     FA=SET     63   A
FA=SET            BX7    X7+X4                                              FA=SET     64   A
FA=SET            SA7    A4          LIMIT                                  FA=SET     65   A
FA=SET            SA6    A6-B1       IN                                     FA=SET     66   A
FA=SET            BX7    X1+X6       24/0, 18/PASS 1 LEN, 18/PASS 1 FWA     FA=SET     67   A
FA=SET            SA1    A6-B1       FET+1                                  FA=SET     68   A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=SET

```
FA=SET            BX1    X1*X3       SAVE FET LENGTH ETC.                          FA=SET      69    A
FA=SET            BX6    X6+X1                                                     FA=SET      70    A
FA=SET            SA6    A1          FIRST                                         FA=SET      71    A
FA=SET            SA7    A7+B1       CBSET                                         FA=SET      72    A
FA=SET            JP     FA=SET      EXIT                                          FA=SET      73    A
FA=SET                                                                             FA=SET      74    A
FA=SET    #RM     ELSE                                                             FA=SET      75    A
FA=SET                                                                             FA=SET      76    A
FA=SET   **       FA=SET - SET FILE BUFFER ADDRESSES (RECORD MANAGER I/O).         FA=SET      77    A
FA=SET   *                                                                         FA=SET      78    A
FA=SET   *        INITIALIZES HOLDING BUFFER ADDRESSES IN FIT AND PSEUDO-FET.      FA=SET      79    A
FA=SET   *        EITHER THE FWA OR LWA+1 OF THE BUFFER SPACE IS REQUIRED AS AN    FA=SET      80    A
FA=SET   *        INPUT PARAMETER.  THE OTHER ADDRESS WILL BE COMPUTED.  BOTH      FA=SET      81    A
FA=SET   *        ADDRESSES ARE IN REGISTERS ON EXIT.  BUFFER LENGTH IS            FA=SET      82    A
FA=SET   *        OBTAINED FROM THE PSEUDO-FET (I.HBSET).                          FA=SET      83    A
FA=SET   *                                                                         FA=SET      84    A
FA=SET   *                                                                         FA=SET      85    A
FA=SET   *        ENTRY  (X2) = PSEUDO-FET WORD 1                                  FA=SET      86    A
FA=SET   *               (A2) = PSEUDO-FET ADDRESS                                 FA=SET      87    A
FA=SET   *               (B1) = 1                                                  FA=SET      88    A
FA=SET   *               (B2) = BUFFER FWA OR 0                                    FA=SET      89    A
FA=SET   *               (B3) = 0 OR BUFFER LWA+1                                  FA=SET      90    A
FA=SET   *                                                                         FA=SET      91    A
FA=SET   *        EXIT   (X2) = PSEUDO-FET WORD 1                                  FA=SET      92    A
FA=SET   *               (A2) = PSEUDO-FET ADDRESS                                 FA=SET      93    A
FA=SET   *               (B1) = 1                                                  FA=SET      94    A
FA=SET   *               (B2) = BUFFER FWA                                         FA=SET      95    A
FA=SET   *               (B3) = BUFFER LWA+1                                       FA=SET      96    A
FA=SET   *                                                                         FA=SET      97    A
FA=SET   *        USES   X - 1, 6, 7                                               FA=SET      98    A
FA=SET   *               A - 1, 6, 7                                               FA=SET      99    A
FA=SET   *               B - 2, 3, 4                                               FA=SET     100    A
FA=SET   *                                                                         FA=SET     101    A
FA=SET   *        SAVES  (A5), (X5)                                                FA=SET     102    A
FA=SET   *                                                                         FA=SET     103    A
FA=SET   *        CALLS  SAR, STORE                                                FA=SET     104    A
FA=SET                                                                             FA=SET     105    A
FA=SET                                                                             FA=SET     106    A
FA=SET    FA=SET  JP     *+4S15      ** ENTRY/EXIT **                              FA=SET     107    A
FA=SET                                                                             FA=SET     108    A
FA=SET   *        INITIALIZE -FILE SPILLED- FLAG TO -UNCONDITIONAL FLUSH-          FA=SET     109    A
FA=SET   *        FLAG VALUE.  THIS ALLOWS SMALL FILES TO BE FLUSHED TO DISK,      FA=SET     110    A
FA=SET   *        EVEN THOUGH THE RELATED HOLDING BUFFER MAY HAVE NEVER BEEN       FA=SET     111    A
FA=SET   *        COMPLETELY FILLED.                                               FA=SET     112    A
FA=SET                                                                             FA=SET     113    A
FA=SET            MX1    1                                                         FA=SET     114    A
FA=SET            LX6    X2,B1                                                     FA=SET     115    A
FA=SET            LX1    -59D+58D                                                  FA=SET     116    A
FA=SET            BX7    -X1*X2      CLEAR OLD SPILL FLAG                          FA=SET     117    A
FA=SET            BX6    X1*X6       EXTRACT FLUSH FLAG                            FA=SET     118    A
FA=SET            BX7    X6+X7                                                     FA=SET     119    A
FA=SET            SA7    A2          RESET SPILL FLAG = FLUSH FLAG                 FA=SET     120    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=SET

```
FA=SET                                                                  FA=SET      121     A
FA=SET   *         DETERMINE IF FWA-UP OR LWA-DOWN MODE.                FA=SET      122     A
FA=SET                                                                  FA=SET      123     A
FA=SET             SB4    B1          (B4) = +1 = MODE SWITCH TO FWA-UP  FA=SET      124     A
FA=SET             NZ     B2,SET2     IF (B2) .NZ., MODE IS FWA-UP       FA=SET      125     A
FA=SET             SB4    -B1         (B4) = -1 = MODE SWITCH TO LWA-DOWN FA=SET     126     A
FA=SET                                                                  FA=SET      127     A
FA=SET    SET2     PL     X2,SET3     IF FILE HAS NO HOLDING BUFFER      FA=SET      128     A
FA=SET             SA1    A2+I.HBSET  (X1) = HOLDING BUFFER RESET PARAMS FA=SET      129     A
FA=SET             RJ     SAR                                            FA=SET      130     A
FA=SET             BX7    X1+X6                                          FA=SET      131     A
FA=SET             SA6    A2+B1       HOLD BUF FWA   TO (P-FET+I.FIRST)  FA=SET      132     A
FA=SET             SA7    A1          ADD FWA TO RESET PARAM WORD        FA=SET      133     A
FA=SET             SA6    A6+B1                      TO (P-FET+I.IN)     FA=SET      134     A
FA=SET             SX7    B3                                             FA=SET      135     A
FA=SET             SA6    A6+B1                      TO (P-FET+I.OUT)    FA=SET      136     A
FA=SET             SA7    A6+B1       HOLD BUF LWA+1 TO (P-FET+I.LIMIT)  FA=SET      137     A
FA=SET             STORE  X2,WSA=X6,X1                                   FA=SET      138     A
FA=SET             SX7    B3-B2       (X7) = HOLDING BUFFER LENGTH (WORDS) FA=SET    139     A
FA=SET             IX6    X7+X7                                          FA=SET      140     A
FA=SET             LX7    3                                              FA=SET      141     A
FA=SET             IX6    X6+X7       (X6) = LENGTH (CHARACTERS)         FA=SET      142     A
FA=SET             STORE  X2,PTL=X6,X1                                   FA=SET      143     A
FA=SET             JP     FA=SET      EXIT                               FA=SET      144     A
FA=SET                                                                  FA=SET      145     A
FA=SET    SET3     MX1    0           DUMMY LENGTH = 0 TO FORCE FWA = LWA+1 FA=SET   146     A
FA=SET             RJ     SAR                                            FA=SET      147     A
FA=SET             JP     FA=SET      EXIT                               FA=SET      148     A
FA=SET    SAR      SPACE  4,8                                            FA=SET      149     A
FA=SET   **        SAR - SET ADDRESSES IN REGISTERS.                    FA=SET      150     A
FA=SET   *                                                              FA=SET      151     A
FA=SET   *                                                              FA=SET      152     A
FA=SET   *         ENTRY  (X1) = LEN * 1S18                             FA=SET      153     A
FA=SET   *                (B2) = FWA OR 0                               FA=SET      154     A
FA=SET   *                (B3) = 0 OR LWA+1                             FA=SET      155     A
FA=SET   *                (B4) = +1 IF FWA-UP MODE AND (B2) = FWA       FA=SET      156     A
FA=SET   *                     = -1 IF LWA-DOWN MODE AND (B3) = LWA+1   FA=SET      157     A
FA=SET   *                                                              FA=SET      158     A
FA=SET   *         EXIT   (X1) = LEN * 1S18  (LOWER 18 BITS CLEAR)      FA=SET      159     A
FA=SET   *                (X6) = FWA                                    FA=SET      160     A
FA=SET   *                (B2) = FWA                                    FA=SET      161     A
FA=SET   *                (B3) = LWA+1                                  FA=SET      162     A
FA=SET   *                                                              FA=SET      163     A
FA=SET   *         USES   X - 1, 6                                      FA=SET      164     A
FA=SET   *                A - NONE                                      FA=SET      165     A
FA=SET   *                B - 2, 3                                      FA=SET      166     A
FA=SET   *                                                              FA=SET      167     A
FA=SET   *         CALLS  NONE                                          FA=SET      168     A
FA=SET                                                                  FA=SET      169     A
FA=SET                                                                  FA=SET      170     A
FA=SET    SAR2     BX6    -X1         (X6) = - BUFFER LENGTH             FA=SET      171     A
FA=SET             LX1    18D                                            FA=SET      172     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=SET

      FA=SET              SB2    X6+B3       (B2) = BUFFER FWA                                  FA=SET        173     A
      FA=SET              SX6    X6+B3                                                          FA=SET        174     A
      FA=SET                                                                                    FA=SET        175     A
      FA=SET    SAR       JP     *+4S15      ** ENTRY/EXIT **                                   FA=SET        176     A
      FA=SET              AX1    18D                                                            FA=SET        177     A
      FA=SET              MI     B4,SAR2     IF LWA-DOWN MODE                                   FA=SET        178     A
      FA=SET              SB3    X1+B2       (B3) = BUFFER LWA+1                                FA=SET        179     A
      FA=SET              LX1    18D                                                            FA=SET        180     A
      FA=SET              SX6    B2                                                             FA=SET        181     A
      FA=SET              EQ     SAR         EXIT                                               FA=SET        182     A
      FA=SET              SPACE  4                                                              FA=SET        183     A
      FA=SET    #RM       ENDIF                                                                 FA=SET        184     A
      FA=SET                                                                                    FA=SET        185     A
      FA=SET                                                                                    FA=SET        186     A
      FA=SET              IF     -DEF,QUAL$,2                                                   FA=SET        187     A
      FA=SET              QUAL   *                                                              FA=SET        188     A
      FA=SET    FA=SET    EQU    /FA=SET/FA=SET                                                 FA=SET        189     A
      FA=SET                                                                                    FA=SET        190     A
      FA=SET              IF     -ABS,FA=SET,1                                                  FA=SET        191     A
      FA=SET              ENTRY  FA=SET                                                         FA=SET        192     A
      FA=SET                                                                                    FA=SET        193     A
      FA=SET              ENDX                                                                  FA=SET        194     A

           SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - FA=SET

                IDENTIFIER                TOTAL    ACTIVE

                FA=SET                      194       194

                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=WTH

      FA=WTH    *COMDECK FA=WTH                                                                 FA=WTH          1     A
      FA=WTH              CTEXT  FA=WTH - WRITE CODED LINE, -H- FORMAT                          FA=WTH          2     A
      FA=WTH     WTH      SPACE  4                                                              FA=WTH          3     A
      FA=WTH              IF     -DEF,QUAL$,1                                                   FA=WTH          4     A
      FA=WTH              QUAL   FA=WTH                                                         FA=WTH          5     A
      FA=WTH                                                                                    FA=WTH          6     A
      FA=WTH                                                                                    FA=WTH          7     A
      FA=WTH    **        FA=WTH - WRITE CODED LINE TO FILE, IN -H- FORMAT.                     FA=WTH          8     A
      FA=WTH    *                                                                               FA=WTH          9     A
      FA=WTH    *               *FA=WTH* PARALLELS ITS KRONOS MODEL, *COMCWTH*.                 FA=WTH         10     A
      FA=WTH    *         LINE LENGTH IN WORDS MUST BE PROVIDED IN (B7) ON ENTRY.               FA=WTH         11     A
      FA=WTH    *         IF (B7) CONTAINS ZERO, NO TRANSFER WILL BE MADE.                      FA=WTH         12     A
      FA=WTH    *                                                                               FA=WTH         13     A
      FA=WTH    *                                                                               FA=WTH         14     A
      FA=WTH    *         ENTRY  (X2) = FIT ADDRESS                                             FA=WTH         15     A
      FA=WTH    *                (B1) = 1                                                       FA=WTH         16     A
      FA=WTH    *                (B6) = LINE ADDRESS                                            FA=WTH         17     A
      FA=WTH    *                (B7) = LINE LENGTH IN WORDS, OR ZERO - SEE ABOVE               FA=WTH         18     A
      FA=WTH    *                                                                               FA=WTH         19     A
      FA=WTH    *         EXIT   (X2) = FIT ADDRESS                                             FA=WTH         20     A
      FA=WTH    *                (B1) = 1                                                       FA=WTH         21     A

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=WTH

```
FA=WTH   *                                                                  FA=WTH     22   A
FA=WTH   *         USES   X - 1, 3, 4, 6, 7                                 FA=WTH     23   A
FA=WTH   *                A - 1, 4                                          FA=WTH     24   A
FA=WTH   *                B - 2, 7                                          FA=WTH     25   A
FA=WTH   *                                                                  FA=WTH     26   A
FA=WTH   *         CALLS  PUT                                               FA=WTH     27   A
FA=WTH                                                                      FA=WTH     28   A
FA=WTH                                                                      FA=WTH     29   A
FA=WTH    FA=WTH   JP     *+4S15      ** ENTRY/EXIT **                      FA=WTH     30   A
FA=WTH             LE     B7,B0,FA=WTH IF NO LENGTH, NO TRANSFER            FA=WTH     31   A
FA=WTH                                                                      FA=WTH     32   A
FA=WTH   *         TEST FOR NO TRAILING BLANKS                              FA=WTH     33   A
FA=WTH                                                                      FA=WTH     34   A
FA=WTH             SA1    B6+B7                                             FA=WTH     35   A
FA=WTH             MX4    48                                                FA=WTH     36   A
FA=WTH             SA1    A1-B1       FETCH LAST WORD                       FA=WTH     37   A
FA=WTH             BX4    -X4*X1                                            FA=WTH     38   A
FA=WTH             ZR     X4,WTH2     IF ZERO BYTE TERMINATION              FA=WTH     39   A
FA=WTH             SA4    WTHB        =1H                                   FA=WTH     40   A
FA=WTH             IX1    X1-X4                                             FA=WTH     41   A
FA=WTH             NZ     X1,WTH2     IF NO TRAILING BLANK WORDS            FA=WTH     42   A
FA=WTH                                                                      FA=WTH     43   A
FA=WTH   *         DELETE TRAILING BLANK WORDS.                             FA=WTH     44   A
FA=WTH                                                                      FA=WTH     45   A
FA=WTH             SA1    B6+B7       PRESET (A1)                           FA=WTH     46   A
FA=WTH             SB7    B7+B1                                             FA=WTH     47   A
FA=WTH                                                                      FA=WTH     48   A
FA=WTH    WTH1     SA1    A1-B1                                             FA=WTH     49   A
FA=WTH             IX6    X1-X4                                             FA=WTH     50   A
FA=WTH             SB7    B7-B1                                             FA=WTH     51   A
FA=WTH             EQ     B7,B1,WTH2                                        FA=WTH     52   A
FA=WTH             ZR     X6,WTH1                                           FA=WTH     53   A
FA=WTH                                                                      FA=WTH     54   A
FA=WTH   *         DETERMINE LINE LENGTH (CHARACTERS).                      FA=WTH     55   A
FA=WTH                                                                      FA=WTH     56   A
FA=WTH    WTH2     SB2    B7-B1                                             FA=WTH     57   A
FA=WTH             SX6    B7                                                FA=WTH     58   A
FA=WTH             SA1    B6+B2       (X1) = LAST WORD OF LINE              FA=WTH     59   A
FA=WTH             IX7    X6+X6                                             FA=WTH     60   A
FA=WTH             LX6    3                                                 FA=WTH     61   A
FA=WTH             SA4    WTHA        (X4) = CHARACTER BYTE MASK            FA=WTH     62   A
FA=WTH             IX7    X6+X7       (X7) = LINE LENGTH (CHARACTERS)       FA=WTH     63   A
FA=WTH                                                                      FA=WTH     64   A
FA=WTH   *         LOCATE AND COUNT LINE TERMINATING ZERO BYTES.  ADJUST LINE FA=WTH  65   A
FA=WTH   *         LENGTH (IN CHARACTERS) ACCORDINGLY.                      FA=WTH     66   A
FA=WTH                                                                      FA=WTH     67   A
FA=WTH             CX6    X1          MINUS ZERO PROTECT                    FA=WTH     68   A
FA=WTH             MX3    -1                                                FA=WTH     69   A
FA=WTH             NZ     X6,WTH3     IF LAST WORD NOT NULL                 FA=WTH     70   A
FA=WTH             SA1    A1-1        (X1) = LAST NON-NULL WORD OF LINE     FA=WTH     71   A
FA=WTH             SX7    X7-10D                                            FA=WTH     72   A
FA=WTH                                                                      FA=WTH     73   A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=WTH

FA=WTH    WTH3     IX6    X1+X3       BORROW PROPAGATES LEFT THRU ALL ZERO BYTES          FA=WTH        74     A
FA=WTH             NO                                                                      FA=WTH        75     A
FA=WTH             BX1    -X1*X6                                                           FA=WTH        76     A
FA=WTH             SX3    B6          (X3) = LINE ADDRESS                                  FA=WTH        77     A
FA=WTH             BX6    X4*X1       RETAIN *40* IN EACH NULL CHAR BYTE POSITION          FA=WTH        78     A
FA=WTH             NO                                                                      FA=WTH        79     A
FA=WTH             CX1    X6          (X1) = NR OF NULL CHARACTERS                         FA=WTH        80     A
FA=WTH             IX4    X7-X1       (X4) = LINE LENGTH (NON-NULL CHARACTERS)             FA=WTH        81     A
FA=WTH             PUT    X2,X3,X4                                                         FA=WTH        82     A
FA=WTH             JP     FA=WTH      EXIT                                                 FA=WTH        83     A
FA=WTH                                                                                     FA=WTH        84     A
FA=WTH    WTHA     DATA   40404040404040404040B                                            FA=WTH        85     A
FA=WTH    WTHB     DATA   1H                                                               FA=WTH        86     A
FA=WTH                                                                                     FA=WTH        87     A
FA=WTH             IF     -DEF,QUAL$,2                                                     FA=WTH        88     A
FA=WTH             QUAL   *                                                                FA=WTH        89     A
FA=WTH    FA=WTH   EQU    /FA=WTH/FA=WTH                                                   FA=WTH        90     A
FA=WTH                                                                                     FA=WTH        91     A
FA=WTH             IF     -ABS,FA=WTH,1                                                    FA=WTH        92     A
FA=WTH             ENTRY  FA=WTH                                                           FA=WTH        93     A
FA=WTH                                                                                     FA=WTH        94     A
FA=WTH             ENDX                                                                    FA=WTH        95     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - FA=WTH

          IDENTIFIER                TOTAL    ACTIVE

          FA=WTH                       95        95

          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=WTW

FA=WTW    *COMDECK,FA=WTW                                                                  FA=WTW         1     A
FA=WTW             CTEXT  FA=WTW - WRITE WORDS FROM WORKING BUFFER                         FA=WTW         2     A
FA=WTW     WTW     SPACE  4                                                                FA=WTW         3     A
FA=WTW             IF     -DEF,QUAL$,1                                                     FA=WTW         4     A
FA=WTW             QUAL   FA=WTW                                                           FA=WTW         5     A
FA=WTW                                                                                     FA=WTW         6     A
FA=WTW                                                                                     FA=WTW         7     A
FA=WTW    **       FA=WTW - WRITE WORDS FROM WORKING BUFFER TO SEQUENTIAL FILE.            FA=WTW         8     A
FA=WTW    *                                                                                FA=WTW         9     A
FA=WTW    *                                                                                FA=WTW        10     A
FA=WTW    *        ENTRY  (X2) = PSEUDO-FET WORD 1                                         FA=WTW        11     A
FA=WTW    *               (A2) = PSEUDO-FET ADDRESS                                        FA=WTW        12     A
FA=WTW    *               (B1) = 1                                                         FA=WTW        13     A
FA=WTW    *               (B6) = WSA ADDRESS                                               FA=WTW        14     A
FA=WTW    *               (B7) = WSA LENGTH (WORDS)                                        FA=WTW        15     A
FA=WTW    *                                                                                FA=WTW        16     A
FA=WTW    *        EXIT   (X2) = PSEUDO-FET WORD 1                                         FA=WTW        17     A
FA=WTW    *               (A2) = PSEUDO-FET ADDRESS                                        FA=WTW        18     A
FA=WTW    *               (B1) = 1                                                         FA=WTW        19     A
FA=WTW    *                                                                                FA=WTW        20     A
FA=WTW    *        USES   X - 1, 2, 3, 4, 6                                                FA=WTW        21     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=WTW

```
FA=WTW   *                A - 1, 2, 3, 4, 6                                  FA=WTW    22    A
FA=WTW   *                B - 2, 3, 6, 7                                     FA=WTW    23    A
FA=WTW   *                                                                   FA=WTW    24    A
FA=WTW   *         CALLS  MOVE, PUTW                                         FA=WTW    25    A
FA=WTW                                                                       FA=WTW    26    A
FA=WTW    FA=WTWA  BSS    1           STORAGE FOR (X5)                       FA=WTW    27    A
FA=WTW    FA=WTWB  BSS    1           STORAGE FOR (A5)                       FA=WTW    28    A
FA=WTW                                                                       FA=WTW    29    A
FA=WTW                                                                       FA=WTW    30    A
FA=WTW    FA=WTW   JP     *+4S15      ** ENTRY/EXIT **                       FA=WTW    31    A
FA=WTW                                                                       FA=WTW    32    A
FA=WTW             IFNE   TEST,0,1                                           FA=WTW    33    A
FA=WTW             LT     B7,B0,*+4S15  IF LENGTH ERROR, ABORT JOB           FA=WTW    34    A
FA=WTW                                                                       FA=WTW    35    A
FA=WTW             ZR     B7,FA=WTW   IF ZERO LENGTH WRITE                   FA=WTW    36    A
FA=WTW    WTW2     SA3    A2+I.IN     (X3) = HOLDING BUFFER -IN- POINTER     FA=WTW    37    A
FA=WTW             SA4    A2+I.LIMIT  (X4) = HOLDING BUFFER -LIMIT- POINTER  FA=WTW    38    A
FA=WTW             SX6    X3+B7       UPDATED -IN-                           FA=WTW    39    A
FA=WTW             IX1    X6-X4       DATA LENGTH .MI. EMPTY BUFFER SPACE    FA=WTW    40    A
FA=WTW             MI     X1,WTW4     IF DATA WILL NOT OVERFLOW BUFFER       FA=WTW    41    A
FA=WTW                                                                       FA=WTW    42    A
FA=WTW   *         FILL HOLDING BUFFER.  CALL RECORD MANAGER TO WRITE IT TO FILE.  FA=WTW  43  A
FA=WTW                                                                       FA=WTW    44    A
FA=WTW             MX6    1                                                  FA=WTW    45    A
FA=WTW             SB2    A2          (B2) = PSEUDO-FET ADDRESS              FA=WTW    46    A
FA=WTW             LX6    -59D+58D                                           FA=WTW    47    A
FA=WTW             SB3    X1          (B3) = LENGTH OF EXCESS DATA           FA=WTW    48    A
FA=WTW             BX6    X2+X6                                              FA=WTW    49    A
FA=WTW             IX1    X4-X3                                              FA=WTW    50    A
FA=WTW             SX2    B6                                                 FA=WTW    51    A
FA=WTW             SA6    A2          FILE SPILLED FLAG .ON.                 FA=WTW    52    A
FA=WTW             SB6    B6+X1       (B6) = FWA EXCESS DATA                 FA=WTW    53    A
FA=WTW             ZR     X1,WTW3     IF ZERO LENGTH MOVE                    FA=WTW    54    A
FA=WTW             MOVE   X1,X2,X3                                           FA=WTW    55    A
FA=WTW    WTW3     SA2    B2          (X2) = FILE CONTROL WORD               FA=WTW    56    A
FA=WTW                                                                       FA=WTW    57    A
FA=WTW   *         NOTE,  ***  COLOSSAL KLUGE  ***.                          FA=WTW    58    A
FA=WTW   *                                                                   FA=WTW    59    A
FA=WTW   *         A5 AND X5 SAVED FOR 7RM *FETCH* MACRO. THIS CODE IS TO    FA=WTW    60    A
FA=WTW   *         BE REMOVED AT THE EARLIEST OPPORTUNITY TO BE SUBSTITUTED  FA=WTW    61    A
FA=WTW   *         BY A MORE *CHIC* METHOD.                                  FA=WTW    62    A
FA=WTW                                                                       FA=WTW    63    A
FA=WTW             BX6    X5                                                 FA=WTW    64    A
FA=WTW             SA6    FA=WTWA     SAVE (X5)                              FA=WTW    65    A
FA=WTW             SX6    A5                                                 FA=WTW    66    A
FA=WTW             SA6    FA=WTWB     SAVE (A5)                              FA=WTW    67    A
FA=WTW                                                                       FA=WTW    68    A
FA=WTW             FETCH  X2,RT,X4,X1,X3,X6                                  FA=WTW    69    A
FA=WTW                                                                       FA=WTW    70    A
FA=WTW             SA3    FA=WTWB                                            FA=WTW    71    A
FA=WTW             SA5    X3          RESTORE (A5)                           FA=WTW    72    A
FA=WTW             SA3    FA=WTWA                                            FA=WTW    73    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FA=WTW

```
FA=WTW            BX5    X3          RESTORE (X5)                                  FA=WTW      74    A
FA=WTW                                                                             FA=WTW      75    A
FA=WTW   *        END OF KLUGE.                                                    FA=WTW      76    A
FA=WTW                                                                             FA=WTW      77    A
FA=WTW            SA3    B2+I.LIMIT  (X3) = -LIMIT-                                FA=WTW      78    A
FA=WTW            SA1    B2+B1       (X1) = -FIRST-                                FA=WTW      79    A
FA=WTW            IX3    X3-X1       (X3) = HOLDING BUFFER LENGTH                  FA=WTW      80    A
FA=WTW            BX6    X1                                                        FA=WTW      81    A
FA=WTW            SA6    A1+B1       RESET -IN- = -FIRST-                          FA=WTW      82    A
FA=WTW            SB7    X4-#ST#                                                   FA=WTW      83    A
FA=WTW            ZR     B7,WTW3A    IF S TYPE RECORD                              FA=WTW      84    A
FA=WTW            PUTW   X2,X1,X3    WRITE HOLDING BUFFER TO FILE                  FA=WTW      85    A
FA=WTW            SB7    B3          (B7) = LENGTH OF UNTRANSFERRED DATA           FA=WTW      86    A
FA=WTW            JP     WTW2        CONTINUE FILLING HOLDING BUFFER               FA=WTW      87    A
FA=WTW                                                                             FA=WTW      88    A
FA=WTW    WTW3A   PUTWP  X2,X1,X3                                                  FA=WTW      89    A
FA=WTW            SB7    B3          (B7) = LENGTH OF UNTRANSFERRED DATA           FA=WTW      90    A
FA=WTW            JP     WTW2        CONTINUE FILLING HOLDING BUFFER               FA=WTW      91    A
FA=WTW                                                                             FA=WTW      92    A
FA=WTW   *        MOVE DATA TO HOLDING BUFFER AND EXIT.                            FA=WTW      93    A
FA=WTW                                                                             FA=WTW      94    A
FA=WTW    WTW4    SB2    A2          TEMP SAVE PSEUDO-FET ADDRESS                  FA=WTW      95    A
FA=WTW            SX1    B7          WSA LENGTH                                    FA=WTW      96    A
FA=WTW            SX2    B6          WSA ADDRESS                                   FA=WTW      97    A
FA=WTW            SA6    A3          UPDATE -IN-                                   FA=WTW      98    A
FA=WTW            MOVE   X1,X2,X3                                                  FA=WTW      99    A
FA=WTW            SA2    B2          RESTORE (A2), (X2) FOR EXIT                   FA=WTW     100    A
FA=WTW            JP     FA=WTW      EXIT                                          FA=WTW     101    A
FA=WTW                                                                             FA=WTW     102    A
FA=WTW                                                                             FA=WTW     103    A
FA=WTW            IF     -DEF,QUAL$,2                                              FA=WTW     104    A
FA=WTW            QUAL   *                                                         FA=WTW     105    A
FA=WTW    FA=WTW  EQU    /FA=WTW/FA=WTW                                            FA=WTW     106    A
FA=WTW                                                                             FA=WTW     107    A
FA=WTW            IF     -ABS,FA=WTW,1                                             FA=WTW     108    A
FA=WTW            ENTRY  FA=WTW                                                    FA=WTW     109    A
FA=WTW                                                                             FA=WTW     110    A
FA=WTW            ENDX                                                             FA=WTW     111    A
```

SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - FA=WTW

IDENTIFIER	TOTAL	ACTIVE
FA=WTW	111	111

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OPRDEFS

```
OPRDEFS  *COMDECK  OPRDEFS                                                         OPRDEFS       1    A
OPRDEFS   OPRDEFS  CTEXT  OPRDEFS - CCG *IL* INSTRUCTION DEFINITIONS               OPRDEFS       2    A
OPRDEFS  **        OPRDEFS - DEFINE THE PROPERTIES OF THE *IL* INSTRUCTIONS.       OPRDEFS       3    A
OPRDEFS   OPRDEFS  SPACE  2                                                        OPRDEFS       4    A
OPRDEFS  **        GETARG - GET ARGUMENT STARTING IN COLUMN N FROM STRING SS       OPRDEFS       5    A
```

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OPRDEFS

```
OPRDEFS  *          M IS MAXIMUM STRING LENGTH                                  OPRDEFS      6     A
OPRDEFS  *          NOTE: S IS THE NULL MICRO IF THE ARG HAS LEADING BLANKS     OPRDEFS      7     A
OPRDEFS                                                                         OPRDEFS      8     A
OPRDEFS             MACRO  GETARG,S,N,M,SS                                      OPRDEFS      9     A
OPRDEFS   S         MICRO  N-15,M,$SS$                                          OPRDEFS     10     A
OPRDEFS   S         MICRO  1,, "S"                                              OPRDEFS     11     A
OPRDEFS   GETARG    ENDM                                                        OPRDEFS     12     A
OPRDEFS                                                                         OPRDEFS     13     A
OPRDEFS  **         V= - SCAN FIRST LIST AND SET RESULT TO VALUE IN SECOND LIST OPRDEFS     14     A
OPRDEFS                                                                         OPRDEFS     15     A
OPRDEFS             MACRO  V=,V,S,L1,L2                                         OPRDEFS     16     A
OPRDEFS   V         SET    -1                                                   OPRDEFS     17     A
OPRDEFS             ECHO   3,P1=(L1),P2=(L2)                                    OPRDEFS     18     A
OPRDEFS             IFC    EQ,/S/P1/,2                                          OPRDEFS     19     A
OPRDEFS   V         SET    P2                                                   OPRDEFS     20     A
OPRDEFS             STOPDUP                                                     OPRDEFS     21     A
OPRDEFS             ERRMI  V           ILLEGAL MACRO PARAM - S                  OPRDEFS     22     A
OPRDEFS   V=        ENDM                                                        OPRDEFS     23     A
OPRDEFS                                                                         OPRDEFS     24     A
OPRDEFS  **         FUNIT - DEFINE FUNCTIONAL UNIT CODES FOR *OPR* MACRO        OPRDEFS     25     A
OPRDEFS                                                                         OPRDEFS     26     A
OPRDEFS             MACRO  FUNIT,N,F66,F76                                      OPRDEFS     27     A
OPRDEFS   M         IFEQ   .CPU,76,2                                            OPRDEFS     28     A
OPRDEFS   N         EQU    F76 F66           DEFAULT TO 6600 UNIT              OPRDEFS     29     A
OPRDEFS   M         SKIP   1                                                    OPRDEFS     30     A
OPRDEFS   N         EQU    F66                                                  OPRDEFS     31     A
OPRDEFS   FUNIT     ENDM                                                        OPRDEFS     32     A
OPRDEFS             SPACE  2                                                    OPRDEFS     33     A
OPRDEFS  **         FUNCTIONAL UNIT CODES FOR 6600/7600                         OPRDEFS     34     A
OPRDEFS  *                                                                      OPRDEFS     35     A
OPRDEFS  * UNIT     FUNIT  66,76 UNIT NUMBER                                    OPRDEFS     36     A
OPRDEFS                                                                         OPRDEFS     37     A
OPRDEFS             QUAL   FUNIT                                                OPRDEFS     38     A
OPRDEFS  *PSEUDO    FUNIT  0                                                    OPRDEFS     39     A
OPRDEFS   BRANCH    FUNIT  1                                                    OPRDEFS     40     A
OPRDEFS   BOOL      FUNIT  2                                                    OPRDEFS     41     A
OPRDEFS   SHIFT     FUNIT  3                                                    OPRDEFS     42     A
OPRDEFS   IADD      FUNIT  4                                                    OPRDEFS     43     A
OPRDEFS   FADD      FUNIT  5                                                    OPRDEFS     44     A
OPRDEFS   MULT      FUNIT  6                 MULT2 = 7 FOR A 6600               OPRDEFS     45     A
OPRDEFS   DIVIDE    FUNIT  8                                                    OPRDEFS     46     A
OPRDEFS   INC       FUNIT  9                 INC2 = 10 ON A 6600                OPRDEFS     47     A
OPRDEFS   NORM      FUNIT  SHIFT,7                                              OPRDEFS     48     A
OPRDEFS   BJUMP     FUNIT  INC,BRANCH                                           OPRDEFS     49     A
OPRDEFS   XJUMP     FUNIT  IADD,BRANCH                                          OPRDEFS     50     A
OPRDEFS   LCM       FUNIT  10                LCM PSEUDO FUNCT UNIT FOR 7600     OPRDEFS     51     A
OPRDEFS   POPC      FUNIT  DIVIDE,11                                            OPRDEFS     52     A
OPRDEFS   PACK      FUNIT  SHIFT,BOOL                                           OPRDEFS     53     A
OPRDEFS             QUAL   *                                                    OPRDEFS     54     A
OPRDEFS   OPRDEFS   SPACE  3,14                                                 OPRDEFS     55     A
OPRDEFS  **         FORMAT OF THE *OPR* DEFINITION IS -                         OPRDEFS     56     A
OPRDEFS  * NAM      OPR ( TYPE  COMM  USI  T6/T7  UNIT1 UNIT2  LEN K=J  SPEC   B )  OPRDEFS     57          I

            0         1         2         3         4         5         6         7         8
            12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OPRDEFS

```
-CCGA092
OPRDEFS  * NAM     OPR ( TYPE  COMM USI T9/T6/T7 UNIT1 UNIT2  LEN K=J  SPEC   B )        CCGA092      5     A
OPRDEFS  *                                                                               OPRDEFS     58     A
OPRDEFS  *         NAME - MEMONIC NAME OF *IL* INSTRUCTION                               OPRDEFS     59     A
OPRDEFS  *         TYPE - INSTRUCTION TYPE ( I - IV )                                    OPRDEFS     60     A
OPRDEFS  *         COMM - "COMM" IF OPERANDS ARE COMMUTATIVE, ELSE "-"                   OPRDEFS     61     A
OPRDEFS  *         USI  - "USI" IF INSTRUCTION IS UNSAFE TO MOVE BECAUSE IT              OPRDEFS     62     A
OPRDEFS  *                MAY CAUSE AN INTERRUPT ( FLOATING POINT OPERATIONS ).          OPRDEFS     63     A
OPRDEFS  *         T9   -  990 EXECUTION TIME OR "-" FOR SPECIAL INST                    CCGA092      6     A
OPRDEFS  *         T6/T7 - 6600/7600 EXECUTION TIME OR "-" FOR SPECIAL INST              OPRDEFS     64     A
OPRDEFS  *                FOR B-JUMPS THE STACK TIME IS USED.                            OPRDEFS     65     A
OPRDEFS  *                FOR X-JUMPS THE FALL THROUGH TIME IS USED.                     OPRDEFS     66     A
OPRDEFS  *                TIME FOR LAB, AND UNCONDITIONAL JUMPS IS SET AT 31             OPRDEFS     67     A
OPRDEFS  *                TO ALLOW THEM TO APPEAR IN THE MIDDLE OF A SEQUENCE.           OPRDEFS     68     A
OPRDEFS  *         UNIT1 - FUNCTIONAL UNIT WHICH EXECUTES THE INSTRUCTION                OPRDEFS     69     A
OPRDEFS  *         UNIT2 - SECOND FUNCTIONAL UNIT, IF AN ALTERNATE UNIT IS               OPRDEFS     70     A
OPRDEFS  *               AVAILABLE AND DIFFERENT FROM THE FIRST.                         OPRDEFS     71     A
OPRDEFS  *               "FU1" IN THE CASE OF DUPLEXED UNITS ON THE 6600                 OPRDEFS     72     A
OPRDEFS  *                ( INCREMENT AND MULTIPLY ).                                    OPRDEFS     73     A
OPRDEFS  *         LEN  - INSTRUCTION LENGTH IN BITS ( 15,30,60,- )                      OPRDEFS     74     A
OPRDEFS  *         K=J - MISCELLANEOUS INSTRUCTION PROPERTIES                            OPRDEFS     75     A
OPRDEFS  *         SPEC - ANY SPECIAL PROPERTIES OF THE INSTRUCTION                      OPRDEFS     76     A
OPRDEFS  *         B   - BOUNDARY MARKER ( BEGINS OR TERMINATES A BASIC BLOCK )          OPRDEFS     77     A
OPRDEFS  *                                                                               OPRDEFS     78     A
OPRDEFS  * NAM     OPR ( TYPE  COMM  USI  T6/T7  UNIT1 UNIT2  LEN K=J  SPEC   B )        OPRDEFS     79          I
-CCGA092
OPRDEFS  *         PSEUDO INSTRUCTIONS                                                   OPRDEFS     80          I
-CCGA092
OPRDEFS   EOQ      OPR (  IV     -    -    1/1    -     -      -             A )        OPRDEFS     81          I
-CCGA092
OPRDEFS   BOS      OPR (  IV     -    -     -     -     -      -                )        OPRDEFS     82          I
-CCGA092
OPRDEFS   EOS      OPR (  IV     -    -     -     -     -      -             A )        OPRDEFS     83          I
-CCGA092
OPRDEFS   DAR      OPR (  I      -    -     -     -     -      -                )        OPRDEFS     84          I
-CCGA092
OPRDEFS   RS       OPR (  II     -    -     -     -     -      -       REGST    )        OPRDEFS     85          I
-CCGA092
OPRDEFS   DEF      OPR (  II     -    -     -     -     -      -             D )        OPRDEFS     86          I
-CCGA092
OPRDEFS   LAB      OPR (  IV     -    -   31/31   -     -      -             B )        OPRDEFS     87          I
-CCGA092
OPRDEFS   ENT      OPR (  IV     -    -     -     -     -      -             B )        OPRDEFS     88          I
-CCGA092
OPRDEFS  *         REGISTER TO REGISTER MACHINE INSTRUCTIONS                             OPRDEFS     89          I
-CCGA092
OPRDEFS   XMT      OPR (  I      -    -    3/2  BOOL   SHIFT  15  K=J           )        OPRDEFS     90          I
-CCGA092
OPRDEFS   AND      OPR (  I    COMM   -    3/2  BOOL    -     15               )        OPRDEFS     91          I
-CCGA092
OPRDEFS   OR       OPR (  I    COMM   -    3/2  BOOL    -     15               )        OPRDEFS     92          I
-CCGA092

         0        1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OPRDEFS

```
OPRDEFS   XOR      OPR (  I    COMM   -    3/2  BOOL    -     15                 )        OPRDEFS      93          I
         -CCGA092
OPRDEFS   XMTC     OPR (  I      -    -    3/2  BOOL    -     15  K=J            )        OPRDEFS      94          I
         -CCGA092
OPRDEFS   STR      OPR (  I      -    -    3/2  BOOL    -     15                 )        OPRDEFS      95          I
         -CCGA092
OPRDEFS   IMP      OPR (  I      -    -    3/2  BOOL    -     15                 )        OPRDEFS      96          I
         -CCGA092
OPRDEFS   EQV      OPR (  I    COMM   -    3/2  BOOL    -     15                 )        OPRDEFS      97          I
         -CCGA092
OPRDEFS   KLS      OPR (  III    -    -    3/2  SHIFT   -     15  K=J            )        OPRDEFS      98          I
         -CCGA092
OPRDEFS   KRS      OPR (  III    -    -    3/2  SHIFT   -     15  K=J            )        OPRDEFS      99          I
         -CCGA092
OPRDEFS   ILS      OPR (  I      -    -    3/2  SHIFT   -     15                 )        OPRDEFS     100          I
         -CCGA092
OPRDEFS   IRS      OPR (  I      -    -    3/2  SHIFT   -     15                 )        OPRDEFS     101          I
         -CCGA092
OPRDEFS   NR       OPR (  I      -  USI    4/3  NORM    -     15  2R             )        OPRDEFS     102          I
         -CCGA092
OPRDEFS   RNZ      OPR (  I      -  USI    4/3  NORM    -     15  2R             )        OPRDEFS     103          I
         -CCGA092
OPRDEFS   UP       OPR (  I      -    -    3/2  PACK    -     15  2R             )        OPRDEFS     104          I
         -CCGA092
OPRDEFS   PK       OPR (  I      -    -    3/2  PACK    -     15                 )        OPRDEFS     105          I
         -CCGA092
OPRDEFS   FA       OPR (  I    COMM  USI    4/4  FADD    -     15                 )        OPRDEFS     106          I
         -CCGA092
OPRDEFS   FS       OPR (  I      -  USI    4/4  FADD    -     15                 )        OPRDEFS     107          I
         -CCGA092
OPRDEFS   DFA      OPR (  I    COMM  USI    4/4  FADD    -     15                 )        OPRDEFS     108          I
         -CCGA092
OPRDEFS   DFS      OPR (  I      -  USI    4/4  FADD    -     15                 )        OPRDEFS     109          I
         -CCGA092
OPRDEFS   RFA      OPR (  I    COMM  USI    4/4  FADD    -     15                 )        OPRDEFS     110          I
         -CCGA092
OPRDEFS   RFS      OPR (  I      -  USI    4/4  FADD    -     15                 )        OPRDEFS     111          I
         -CCGA092
OPRDEFS   IA       OPR (  I    COMM   -    3/2  IADD    -     15                 )        OPRDEFS     112          I
         -CCGA092
OPRDEFS   IS       OPR (  I      -    -    3/2  IADD    -     15                 )        OPRDEFS     113          I
         -CCGA092
OPRDEFS   FM       OPR (  I    COMM  USI   10/5  MULT   FU1    15                 )        OPRDEFS     114          I
         -CCGA092
OPRDEFS   RFM      OPR (  I    COMM  USI   10/5  MULT   FU1    15                 )        OPRDEFS     115          I
         -CCGA092
OPRDEFS   DFM      OPR (  I    COMM  USI   10/5  MULT   FU1    15                 )        OPRDEFS     116          I
         -CCGA092
OPRDEFS   FMA      OPR (  II     -    -    3/2  SHIFT   -     15  K=J  ZEROP     )        OPRDEFS     117          I
         -CCGA092
OPRDEFS   FD       OPR (  I      -  USI   29/20 DIVIDE  -     15                 )        OPRDEFS     118          I
         -CCGA092

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OPRDEFS

```
OPRDEFS   RFD      OPR (  I      -   USI  29/20 DIVIDE  -     15                )        OPRDEFS      119          I
          -CCGA092
OPRDEFS   NOP      OPR (  III    -    -     -     -     -     -       REGST     )        OPRDEFS      120          I
          -CCGA092
OPRDEFS   CX       OPR (  I      -    -    8/2  POPC    -     15  K=J           )        OPRDEFS      121          I
          -CCGA092
OPRDEFS   *        MEMORY REFERENCE AND SET (LOAD IMMEDIATE ) INSTRUCTIONS                OPRDEFS      122          I
          -CCGA092
OPRDEFS   LD       OPR (  III    -    -    8/8  INC    FU1    30       LOAD      )        OPRDEFS      123          I
          -CCGA092
OPRDEFS   ST       OPR (  III    -    -   10/2  INC    FU1    30       STORE     )        OPRDEFS      124          I
          -CCGA092
OPRDEFS   STT      OPR (  III    -    -    3/2  INC    FU1    30                 )        OPRDEFS      125          I
          -CCGA092
OPRDEFS   PLD      OPR (  I      -    -    8/8  INC    FU1    30       LOAD      )        OPRDEFS      126          I
          -CCGA092
OPRDEFS   PST      OPR (  I      -    -   10/2  INC    FU1    30       STORE     )        OPRDEFS      127          I
          -CCGA092
OPRDEFS   S        OPR (  II     -    -    3/2  INC    FU1    30       ZEROP     )        OPRDEFS      128          I
          -CCGA092
OPRDEFS   SLD      OPR (  I    COMM   -    8/8  INC    FU1    15       LOAD      )        OPRDEFS      129          I
          -CCGA092
OPRDEFS   SST      OPR (  I    COMM   -   10/2  INC    FU1    15       STORE     )        OPRDEFS      130          I
          -CCGA092
OPRDEFS   SA       OPR (  I    COMM   -    3/2  INC    FU1    15                 )        OPRDEFS      131          I
          -CCGA092
OPRDEFS   SDL      OPR (  I      -    -    8/8  INC    FU1    15       LOAD      )        OPRDEFS      132          I
          -CCGA092
OPRDEFS   SDS      OPR (  I      -    -   10/2  INC    FU1    15       STORE     )        OPRDEFS      133          I
          -CCGA092
OPRDEFS   SS       OPR (  I      -    -    3/2  INC    FU1    15                 )        OPRDEFS      134          I
          -CCGA092
OPRDEFS   DRL      OPR (  I      -    -    0/17 LCM     -     15       LOAD2     )        OPRDEFS      135          I
          -CCGA092
OPRDEFS   DWL      OPR (  I      -    -    0/3  LCM     -     15       STORE2    )        OPRDEFS      136          I
          -CCGA092
OPRDEFS   *        CONDITIONAL JUMPS                                                      OPRDEFS      137          I
          -CCGA092
OPRDEFS   RJXJ     OPR (  III    -    -    9/3  XJUMP   -     30       ECJUMP    )        OPRDEFS      138          I
          -CCGA092
OPRDEFS   JPX      OPR (  III    -    -   11/2  XJUMP   -     30       JUMP    C )        OPRDEFS      139          I
          -CCGA092
OPRDEFS   JPBB     OPR (  III    -    -    8/3  BJUMP   -     30       JUMP    C )        OPRDEFS      140          I
          -CCGA092
OPRDEFS   *        UNCONDITIONAL JUMPS                                                    OPRDEFS      141          I
          -CCGA092
OPRDEFS   JIN      OPR (  III    -    -   14/13 INC    FU1    30       UJUMP   A )        OPRDEFS      142          I
          -CCGA092
OPRDEFS   RJ3      OPR (  IV     -    -   31/31 BRANCH  -     30       UJUMP   A )        OPRDEFS      143          I
          -CCGA092
OPRDEFS   RJ6      OPR (  IV     -    -   31/31 BRANCH  -     30       UJUMP   A )        OPRDEFS      144          I
          -CCGA092

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OPRDEFS

```
OPRDEFS   UJP      OPR (  IV     -    -   14/13 BJUMP  FU1    30       UJUMP  A )         OPRDEFS      145          I
          -CCGA092
OPRDEFS   *        SPECIAL INSTRUCTIONS FOR THE OPTIMIZER                                  OPRDEFS      146          I
          -CCGA092
OPRDEFS   ILD      OPR (  III    -    -    8/8  INC    FU1    30       LOAD     )         OPRDEFS      147          I
          -CCGA092
OPRDEFS   TLD      OPR (  III    -    -    8/8  INC    FU1    30       LOAD     )         OPRDEFS      148          I
          -CCGA092
OPRDEFS   TST      OPR (  III    -    -   10/2  INC    FU1    30       STORE    )         OPRDEFS      149          I
          -CCGA092
OPRDEFS   IAZ      OPR (  I    COMM   -    3/2  IADD    -     15                )         OPRDEFS      150          I
          -CCGA092
OPRDEFS   ISZ      OPR (  I      -    -    3/2  IADD    -     15                )         OPRDEFS      151          I
          -CCGA092
OPRDEFS   IM       OPR (  I    COMM   -   10/5  MULT   FU1    15                )         OPRDEFS      152          I
          -CCGA092
OPRDEFS   LDC      OPR (  III    -    -    8/8  INC    FU1    30       LOAD     )         OPRDEFS      153          I
          -CCGA092
OPRDEFS   LDV      OPR (  III    -    -    8/8  INC    FU1    30       LOAD     )         CCG16          1          I
          -CCGA092
OPRDEFS   CLR      OPR (  II     -    -    3/2  SHIFT  BOOL   15  K=J  ZEROP    )         OPRDEFS      154          I
          -CCGA092
OPRDEFS   SXT      OPR (  I      -    -    3/2  BOOL   SHIFT  15  K=J           )         OPRDEFS      155          I
          -CCGA092
OPRDEFS   * NAM    OPR ( TYPE  COMM USI T9/T6/T7 UNIT1 UNIT2  LEN K=J  SPEC   B )         CCGA092        7     A
OPRDEFS   *        PSEUDO INSTRUCTIONS                                                     CCGA092        8     A
OPRDEFS   EOQ      OPR (  IV     -    -  1/ 1/1    -     -      -             A )         CCGA092        9     A
OPRDEFS   BOS      OPR (  IV     -    -     -     -     -      -                )         CCGA092       10     A
OPRDEFS   EOS      OPR (  IV     -    -     -     -     -      -             A )         CCGA092       11     A
OPRDEFS   DAR      OPR (  I      -    -     -     -     -      -                )         CCGA092       12     A
OPRDEFS   RS       OPR (  II     -    -     -     -     -      -       REGST    )         CCGA092       13     A
OPRDEFS   DEF      OPR (  II     -    -     -     -     -      -             D )         CCGA092       14     A
OPRDEFS   LAB      OPR (  IV     -    - 31/31/31   -     -      -             B )         CCGA092       15     A
OPRDEFS   ENT      OPR (  IV     -    -     -     -     -      -             B )         CCGA092       16     A
OPRDEFS   *        REGISTER TO REGISTER MACHINE INSTRUCTIONS                               CCGA092       17     A
OPRDEFS   XMT      OPR (  I      -    -  3/ 3/2  BOOL   SHIFT  15  K=J           )         CCGA092       18     A
OPRDEFS   AND      OPR (  I    COMM   -  3/ 3/2  BOOL    -     15                )         CCGA092       19     A
OPRDEFS   OR       OPR (  I    COMM   -  3/ 3/2  BOOL    -     15                )         CCGA092       20     A
OPRDEFS   XOR      OPR (  I    COMM   -  3/ 3/2  BOOL    -     15                )         CCGA092       21     A
OPRDEFS   XMTC     OPR (  I      -    -  3/ 3/2  BOOL    -     15  K=J           )         CCGA092       22     A
OPRDEFS   STR      OPR (  I      -    -  3/ 3/2  BOOL    -     15                )         CCGA092       23     A
OPRDEFS   IMP      OPR (  I      -    -  3/ 3/2  BOOL    -     15                )         CCGA092       24     A
OPRDEFS   EQV      OPR (  I    COMM   -  3/ 3/2  BOOL    -     15                )         CCGA092       25     A
OPRDEFS   KLS      OPR (  III    -    -  4/ 3/2  SHIFT   -     15  K=J           )         CCGA092       26     A
OPRDEFS   KRS      OPR (  III    -    -  4/ 3/2  SHIFT   -     15  K=J           )         CCGA092       27     A
OPRDEFS   ILS      OPR (  I      -    -  4/ 3/2  SHIFT   -     15                )         CCGA092       28     A
OPRDEFS   IRS      OPR (  I      -    -  4/ 3/2  SHIFT   -     15                )         CCGA092       29     A
OPRDEFS   NR       OPR (  I      -  USI 6/ 4/3  NORM    -     15  2R            )         CCGA092       30     A
OPRDEFS   RNZ      OPR (  I      -  USI 6/ 4/3  NORM    -     15  2R            )         CCGA092       31     A
OPRDEFS   UP       OPR (  I      -    -  3/ 3/2  PACK    -     15  2R            )         CCGA092       32     A
OPRDEFS   PK       OPR (  I      -    -  3/ 3/2  PACK    -     15                )         CCGA092       33     A
OPRDEFS   FA       OPR (  I    COMM USI 7/ 4/4  FADD    -     15                )         CCGA092       34     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OPRDEFS

```
OPRDEFS   FS      OPR (  I      -  USI 7/ 4/4  FADD    -     15                )        CCGA092       35     A
OPRDEFS   DFA     OPR (  I    COMM USI 7/ 4/4  FADD    -     15                )        CCGA092       36     A
OPRDEFS   DFS     OPR (  I      -  USI 7/ 4/4  FADD    -     15                )        CCGA092       37     A
OPRDEFS   RFA     OPR (  I    COMM USI 7/ 4/4  FADD    -     15                )        CCGA092       38     A
OPRDEFS   RFS     OPR (  I      -  USI 7/ 4/4  FADD    -     15                )        CCGA092       39     A
OPRDEFS   IA      OPR (  I    COMM USI 3/ 3/2  IADD    -     15                )        CCGA092       40     A
OPRDEFS   IS      OPR (  I      -   -  3/ 3/2  IADD    -     15                )        CCGA092       41     A
OPRDEFS   FM      OPR (  I    COMM USI 7/10/5  MULT   FU1    15                )        CCGA092       42     A
OPRDEFS   RFM     OPR (  I    COMM USI 7/10/5  MULT   FU1    15                )        CCGA092       43     A
OPRDEFS   DFM     OPR (  I    COMM USI 7/10/5  MULT   FU1    15                )        CCGA092       44     A
OPRDEFS   FMA     OPR (  II     -   -  4/ 3/2  SHIFT   -     15  K=J  ZEROP    )        CCGA092       45     A
OPRDEFS   FD      OPR (  I      -  USI31/29/20 DIVIDE  -     15                )        CCGA092       46     A
OPRDEFS   RFD     OPR (  I      -  USI31/29/20 DIVIDE  -     15                )        CCGA092       47     A
OPRDEFS   NOP     OPR (  III    -   -     -     -     -      -       REGST    )        CCGA092       48     A
OPRDEFS   CX      OPR (  I      -   -  6/ 8/2  POPC    -     15  K=J           )        CCGA092       49     A
OPRDEFS   *       MEMORY REFERENCE AND SET (LOAD IMMEDIATE ) INSTRUCTIONS                CCGA092       50     A
OPRDEFS   LD      OPR (  III    -   -  8/ 8/8  INC    FU1    30       LOAD     )        CCGA092       51     A
OPRDEFS   ST      OPR (  III    -   -  3/10/2  INC    FU1    30       STORE    )        CCGA092       52     A
OPRDEFS   STT     OPR (  III    -   -  3/ 3/2  INC    FU1    30                )        CCGA092       53     A
OPRDEFS   PLD     OPR (  I      -   -  8/ 8/8  INC    FU1    30       LOAD     )        CCGA092       54     A
OPRDEFS   PST     OPR (  I      -   -  3/10/2  INC    FU1    30       STORE    )        CCGA092       55     A
OPRDEFS   S       OPR (  II     -   -  3/ 3/2  INC    FU1    30       ZEROP    )        CCGA092       56     A
OPRDEFS   SLD     OPR (  I    COMM  -  8/ 8/8  INC    FU1    15       LOAD     )        CCGA092       57     A
OPRDEFS   SST     OPR (  I    COMM  -  3/10/2  INC    FU1    15       STORE    )        CCGA092       58     A
OPRDEFS   SA      OPR (  I    COMM  -  3/ 3/2  INC    FU1    15                )        CCGA092       59     A
OPRDEFS   SDL     OPR (  I      -   -  8/ 8/8  INC    FU1    15       LOAD     )        CCGA092       60     A
OPRDEFS   SDS     OPR (  I      -   -  3/10/2  INC    FU1    15       STORE    )        CCGA092       61     A
OPRDEFS   SS      OPR (  I      -   -  3/ 3/2  INC    FU1    15                )        CCGA092       62     A
OPRDEFS   DRL     OPR (  I      -   -  8/ 0/17 LCM     -     15       LOAD2    )        CCGA092       63     A
OPRDEFS   DWL     OPR (  I      -   -  3/ 0/3  LCM     -     15       STORE2   )        CCGA092       64     A
OPRDEFS   *       CONDITIONAL JUMPS                                                      CCGA092       65     A
OPRDEFS   RJXJ    OPR (  III    -   -  5/ 9/3  XJUMP   -     30       ECJUMP   )        CCGA092       66     A
OPRDEFS   JPX     OPR (  III    -   -  5/11/2  XJUMP   -     30       JUMP   C )        CCGA092       67     A
OPRDEFS   JPBB    OPR (  III    -   -  5/ 8/3  BJUMP   -     30       JUMP   C )        CCGA092       68     A
OPRDEFS   *       UNCONDITIONAL JUMPS                                                    CCGA092       69     A
OPRDEFS   JIN     OPR (  III    -   - 29/14/13 INC    FU1    30       UJUMP  A )        CCGA092       70     A
OPRDEFS   RJ3     OPR (  IV     -   - 31/31/31 BRANCH  -     30       UJUMP  A )        CCGA092       71     A
OPRDEFS   RJ6     OPR (  IV     -   - 31/31/31 BRANCH  -     30       UJUMP  A )        CCGA092       72     A
OPRDEFS   UJP     OPR (  IV     -   -  5/14/13 BJUMP  FU1    30       UJUMP  A )        CCGA092       73     A
OPRDEFS   *       SPECIAL INSTRUCTIONS FOR THE OPTIMIZER                                 CCGA092       74     A
OPRDEFS   ILD     OPR (  III    -   -  8/ 8/8  INC    FU1    30       LOAD     )        CCGA092       75     A
OPRDEFS   TLD     OPR (  III    -   -  8/ 8/8  INC    FU1    30       LOAD     )        CCGA092       76     A
OPRDEFS   TST     OPR (  III    -   -  3/10/2  INC    FU1    30       STORE    )        CCGA092       77     A
OPRDEFS   IAZ     OPR (  I    COMM  -  3/ 3/2  IADD    -     15                )        CCGA092       78     A
OPRDEFS   ISZ     OPR (  I      -   -  3/ 3/2  IADD    -     15                )        CCGA092       79     A
OPRDEFS   IM      OPR (  I    COMM  -  7/10/5  MULT   FU1    15                )        CCGA092       80     A
OPRDEFS   LDC     OPR (  III    -   -  8/ 8/8  INC    FU1    30       LOAD     )        CCGA092       81     A
OPRDEFS   LDV     OPR (  III    -   -  8/ 8/8  INC    FU1    30       LOAD     )        CCGA092       82     A
OPRDEFS   CLR     OPR (  II     -   -  4/ 3/2  SHIFT  BOOL   15  K=J  ZEROP    )        CCGA092       83     A
OPRDEFS   SXT     OPR (  I      -   -  3/ 3/2  BOOL   SHIFT  15  K=J           )        CCGA092       84     A
OPRDEFS                                                                                 OPRDEFS      156     A
OPRDEFS   *       DEFINE MICROS FOR THE CONDITIONAL X AND B JUMPS                        OPRDEFS      157     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OPRDEFS

OPRDEFS                                                                    OPRDEFS      158     A
OPRDEFS   BJUMPS   MICRO  1,,/EQ,NE,GE,LT/                                 OPRDEFS      159     A
OPRDEFS   RELOPS   MICRO  1,,/EQ,NE,GT,LT,GE,LE/                           OPRDEFS      160     A
OPRDEFS   XJUMPS   MICRO  1,,/ZR,NZ,PL,MI,IR,OR,DF,ID/                     OPRDEFS      161     A
OPRDEFS   ^XJUMPS  MICRO  1,,/NZ,ZR,MI,PL,OR,IR,ID,DF/                     OPRDEFS      162     A
OPRDEFS                                                                    OPRDEFS      163     A
OPRDEFS            PURGMAC GETARG,FUNIT,V=,OPR                             OPRDEFS      164     A
OPRDEFS            ENDX                                                    OPRDEFS      165     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - OPRDEFS

          IDENTIFIER                TOTAL    ACTIVE

          OPRDEFS                     165        87
          CCG16                         1         0
          CCGA092                      80        80

          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PSODEFS

PSODEFS  *COMDECK  PSODEFS                                                 PSODEFS        1     A
PSODEFS   PSODEFS  CTEXT  PSODEFS - CCG PSEUDO OPERATION DEFINITIONS       PSODEFS        2     A
PSODEFS  **        PSODEFS - PSEUDO OPERATION DEFINITIONS                  PSODEFS        3     A
PSODEFS                                                                    PSODEFS        4     A
PSODEFS  *               INITIAL GROUP                                     PSODEFS        5     A
PSODEFS                                                                    PSODEFS        6     A
PSODEFS   LCC      POD                LOADER CONTROL CARD                  PSODEFS        7     A
PSODEFS   IDENT    POD                IDENT LINE                           PSODEFS        8     A
PSODEFS   TITLE    POD                TITLE LINE                           PSODEFS        9     A
PSODEFS   COMNT    POD                COMMENT DIRECTIVE                    PSODEFS       10     A
PSODEFS   LIB      POD                LDSET LIB=LIBNAME                    PSODEFS       11     A
PSODEFS   USBLK    POD                *USEBLK* LINE, TERMINATE INITIAL GROUP PSODEFS       12     A
PSODEFS                                                                    PSODEFS       13     A
PSODEFS  *               THE BELOW MAY APPEAR IN *MACRO_S* MACROS          PSODEFS       14     A
PSODEFS                                                                    PSODEFS       15     A
PSODEFS   USE      POD                SET NEW *USE* BLOCK                  PSODEFS       16     A
PSODEFS   BSS      POD                RESERVE STORAGE                      PSODEFS       17     A
PSODEFS   CON      POD                DEFINE RELOCATABLE DATA WORD         PSODEFS       18     A
PSODEFS   BSSZ     POD                RESERVE STORAGE AND ZERO IT OUT      PSODEFS       19     A
PSODEFS                                                                    PSODEFS       20     A
PSODEFS  *               THE BELOW MAY NOT APPEAR IN *MACRO_S* MACROS      PSODEFS       21     A
PSODEFS                                                                    PSODEFS       22     A
PSODEFS   BMI      POD                BEGIN *TS* FORMAT MACHINE INST PROCESSING PSODEFS       23     A
PSODEFS   DATA     POD                DEFINE DATA WORDS                    PSODEFS       24     A
PSODEFS   DIS      POD                DEFINE BCD DATA IN *H* FORMAT        PSODEFS       25     A
PSODEFS   HOL      POD                DEFINE HOLLERITH DATA ( H, L, OR R FORMAT ) PSODEFS       26     A
PSODEFS   ORG      POD                SET/RESET *ORG* COUNTER              PSODEFS       27     A
PSODEFS   REPI     POD                REPLICATE DATA                       PSODEFS       28     A
PSODEFS   DCS      POD                DEFINE CHAR STRING AS MICRO NAME     PSODEFS       29     A
PSODEFS   VFDP     POD                CLEAR & ENTER PART WORD OF DATA      PSODEFS       30     A
PSODEFS   APL      POD                DEFINE SUBROUTINE APLIST ENTRY       PSODEFS       31     A
PSODEFS   IOM      POD                DEFINE I/O APLIST ENTRY              PSODEFS       32     A
PSODEFS   SUB      POD                DEFINE SCM/LCM INST AND ADDR SUB WORD PSODEFS       33     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PSODEFS

PSODEFS                                                                          PSODEFS        34     A
PSODEFS  *                THE FOLLOWING PSEUDO ARE USED IN *MACRO_S* ONLY        PSODEFS        35     A
PSODEFS                                                                          PSODEFS        36     A
PSODEFS   AAC      POD                ADD ADDRESS CONSTANT  ( MAY BE CA OF CON. )  PSODEFS        37     A
PSODEFS   ACI      POD                ADD CONSTANT INFO                           PSODEFS        38     A
PSODEFS   ARI      POD                ADD RELOCATABLE INFO                        PSODEFS        39     A
PSODEFS   ASV      POD                ADD STRING VALUE                            PSODEFS        40     A
PSODEFS   BTW      POD                BEGIN TEXT WORD                             PSODEFS        41     A
PSODEFS   ELSE     POD                ELSE                                        PSODEFS        42     A
PSODEFS   IFT      POD                ASSEMBLY TIME IF TEST                       PSODEFS        43     A
PSODEFS   MIC      POD                DEFINE MICRO STRING ( @ 10 CHARS )          PSODEFS        44     A
PSODEFS   SET      POD                DEFINE SET SYMBOL VALUE                     PSODEFS        45     A
PSODEFS                                                                          PSODEFS        46     A
PSODEFS  *                SOME EQUIVALENCES TO THE *IL* OPCODES                  PSODEFS        47     A
PSODEFS                                                                          PSODEFS        48     A
PSODEFS  * END     =      EOQ         END OF INPUT                               PSODEFS        49     A
PSODEFS  * EOB     =      EOS         END OF BUFFER ON INPUT                     PSODEFS        50     A
PSODEFS                                                                          PSODEFS        51     A
PSODEFS            PURGMAC POD                                                   PSODEFS        52     A
PSODEFS            ENDX                                                          PSODEFS        53     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - PSODEFS

          IDENTIFIER                TOTAL    ACTIVE

          PSODEFS                     53        53

          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCGDBGM

CCGDBGM  *COMDECK  CCGDBGM                                                       CCGDBGM         1     A
CCGDBGM            CTEXT  CCGDBGM - CCG DEBUG MACROS                             CCGDBGM         2     A
CCGDBGM                                                                          CCGDBGM         3     A
CCGDBGM  **        PRINT - PRINT THE CONTENTS OF A LIST OF LOCATIONS.            CCGDBGM         4     A
CCGDBGM  *                                                                       CCGDBGM         5     A
CCGDBGM  *                                                                       CCGDBGM         6     A
CCGDBGM  *         PRINT  LAB,FMT,(LIST)                                         CCGDBGM         7     A
CCGDBGM  *                                                                       CCGDBGM         8     A
CCGDBGM  *         LAB - STATEMENT PRINTED IF *LAB* MENTIONED ON A *TRACER* LIST CCGDBGM         9     A
CCGDBGM  *               IF BLANK, THEN UNCONDITIONALLY PRINT.                   CCGDBGM        10     A
CCGDBGM  *         FMT - PARENTHESIED FORTRAN FORMAT                             CCGDBGM        11     A
CCGDBGM  *         LIST - PARENTHESIZED LIST OF NAMES AND OR REGISTERS TO BE     CCGDBGM        12     A
CCGDBGM  *               PRINTED.                                                CCGDBGM        13     A
CCGDBGM  *                                                                       CCGDBGM        14     A
CCGDBGM  *         SAMPLE CALL --                                                CCGDBGM        15     A
CCGDBGM  *         PRINT XXX,(* SIP,IIP =*,2I6),(SIP,IIP)   WILL PRODUCE -       CCGDBGM        16     A
CCGDBGM  *         XXX SIP,IIP = NNNNNN NNNNNN                                   CCGDBGM        17     A
CCGDBGM  *                                                                       CCGDBGM        18     A
CCGDBGM  *         THIS MACRO IS FOR INTERNAL DEBUGGING ONLY ( TEST MODE ).      CCGDBGM        19     A
CCGDBGM                                                                          CCGDBGM        20     A
CCGDBGM                                                                          CCGDBGM        21     A
CCGDBGM            PURGMAC   PRINT                                               CCGDBGM        22     A
CCGDBGM                                                                          CCGDBGM        23     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCGDBGM

```
CCGDBGM   PRINT    MACRO  LAB,FMT,LIST,NS                                  CCGDBGM    24    A
CCGDBGM            LOCAL  APL,LFMT                                         CCGDBGM    25    A
CCGDBGM   .P       IFC    NE,/LAB//,2                                      CCGDBGM    26    A
CCGDBGM   .P       IF     -DEF,)LAB,1                                      CCGDBGM    27    A
CCGDBGM   .P       IF     DEF,/DEBUG/LAB                                   CCGDBGM    28    A
CCGDBGM            USE    DEBUG                                            CCGDBGM    29    A
CCGDBGM   APL      CON    =XOUTPUT#                                        CCGDBGM    30    A
CCGDBGM            CON    LFMT                                             CCGDBGM    31    A
CCGDBGM            IRP    LIST                                             CCGDBGM    32    A
CCGDBGM            IF     REG,LIST,4                                       CCGDBGM    33    A
CCGDBGM   L        MICRO  1,1,/LIST/                                       CCGDBGM    34    A
CCGDBGM   N        MICRO  2,1,/LIST/                                       CCGDBGM    35    A
CCGDBGM            VFD    12/2,30/1S6,18/=XSV="L"+"N"                      CCGDBGM    36    A
CCGDBGM            SKIP   1                                                CCGDBGM    37    A
CCGDBGM            VFD    12/2,30/1S6,18/LIST                              CCGDBGM    38    A
CCGDBGM            IRP                                                     CCGDBGM    39    A
CCGDBGM   LFMT     CON    0                                                CCGDBGM    40    A
CCGDBGM            DIS    ,$(* LAB *,FMT)$                                 CCGDBGM    41    A
CCGDBGM            USE    *                                                CCGDBGM    42    A
CCGDBGM            IFC    NE,/NS/NOSAVE/,1                                 CCGDBGM    43    A
CCGDBGM            RJ     =XSVR=                                           CCGDBGM    44    A
CCGDBGM            SA1    APL                                              CCGDBGM    45    A
CCGDBGM            RJ     =XOUTCI.                                         CCGDBGM    46    A
CCGDBGM            IFC    NE,/NS/NOSAVE/,1                                 CCGDBGM    47    A
CCGDBGM            RJ     =XRSR=                                           CCGDBGM    48    A
CCGDBGM   .P       ENDIF                                                   CCGDBGM    49    A
CCGDBGM   PRINT    ENDM                                                    CCGDBGM    50    A
CCGDBGM   TRACER   SPACE  3,14                                             CCGDBGM    51    A
CCGDBGM  **        TRACER - DEFINE ROUTINES/PHASES TO BE TRACED.           CCGDBGM    52    A
CCGDBGM  *                                                                 CCGDBGM    53    A
CCGDBGM  *         TRACER (RTN1,...,RTNI)                                  CCGDBGM    54    A
CCGDBGM                                                                    CCGDBGM    55    A
CCGDBGM   TRACER   MACRO  P                                                CCGDBGM    56    A
CCGDBGM            QUAL   DEBUG                                            CCGDBGM    57    A
CCGDBGM            IRP    P                                                CCGDBGM    58    A
CCGDBGM   P        EQU    1                                                CCGDBGM    59    A
CCGDBGM            IRP                                                     CCGDBGM    60    A
CCGDBGM            QUAL   *                                                CCGDBGM    61    A
CCGDBGM            ENDM                                                    CCGDBGM    62    A
CCGDBGM   TRACE    SPACE  3                                                CCGDBGM    63    A
CCGDBGM  **        TRACE - CONDITIONALLY SNAP CONTENTS OF SPECIFIED TABLE. CCGDBGM    64    A
CCGDBGM  *         CALL -                                                  CCGDBGM    65    A
CCGDBGM  *         TRACE  LAB,TBL,BLK                                      CCGDBGM    66    A
CCGDBGM  *                                                                 CCGDBGM    67    A
CCGDBGM  *         LAB - SNAP LABEL THAT IS TO BE SPECIFIED ON *TRACER* LIST CCGDBGM  68    A
CCGDBGM  *               TO ACTIVATE THE PRINTOUT.                         CCGDBGM    69    A
CCGDBGM  *         TBL -  IF NOT = RLIST, THEN NAME OF TABLE ( FWA,LEN = O.TBL, CCGDBGM 70  A
CCGDBGM  *               L.TBL ) TO BE PRINTED IN OCTAL FORMAT.            CCGDBGM    71    A
CCGDBGM  *               IF = RLIST , THEN CALL *DMPRLST* TO PRINT THE TABLE. CCGDBGM 72  A
CCGDBGM  *               TABLE PRINTED OUT IS *TXT* IF *BLK* IS NOT SPECIFIED, CCGDBGM 73  A
CCGDBGM  *               ELSE *BLK*.                                       CCGDBGM    74    A
CCGDBGM                                                                    CCGDBGM    75    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCGDBGM

```
CCGDBGM   TRACE    MACRO  RTN,TBL,BLK                                          CCGDBGM      76     A
CCGDBGM   T        IF     DEF,/DEBUG/RTN                                       CCGDBGM      77     A
CCGDBGM            IRP    TBL                                                  CCGDBGM      78     A
CCGDBGM   TB       IFC    NE,/TBL/RLIST/,2                                     CCGDBGM      79     A
CCGDBGM            SNAPT  TBL,TBL-RTN                                          CCGDBGM      80     A
CCGDBGM   TB       ELSE   1                                                    CCGDBGM      81     A
CCGDBGM   BLK      SNAPRL RTN                                                  CCGDBGM      82     A
CCGDBGM            IRP                                                         CCGDBGM      83     A
CCGDBGM   T        ENDIF                                                       CCGDBGM      84     A
CCGDBGM            ENDM                                                        CCGDBGM      85     A
CCGDBGM   SNAPRL   SPACE  3,14                                                 CCGDBGM      86     A
CCGDBGM  **        SNAPRL - INTERPRETIVE DUMP OF AN *IL* INSTRUCTION SEQUENCE  CCGDBGM      87     A
CCGDBGM                                                                        CCGDBGM      88     A
CCGDBGM            PURGMAC  SNAPRL                                             CCGDBGM      89     A
CCGDBGM            MACRO  SNAPRL,BLK,L,LL,UL,INC                               CCGDBGM      90     A
CCGDBGM            LOCAL  X,Y                                                  CCGDBGM      91     A
CCGDBGM            RJ     =XSVR=                                               CCGDBGM      92     A
CCGDBGM            SA1    X                                                    CCGDBGM      93     A
CCGDBGM            IFC    EQ,//BLK/,2                                          CCGDBGM      94     A
CCGDBGM            SA2    =XO$TXT                                              CCGDBGM      95     A
CCGDBGM            ELSE   1                                                    CCGDBGM      96     A
CCGDBGM            SA2    =XO$BLK                                              CCGDBGM      97     A
CCGDBGM            BX6    X2                                                   CCGDBGM      98     A
CCGDBGM            SA6    X+1                                                  CCGDBGM      99     A
CCGDBGM            RJ     =XDMPRLST                                            CCGDBGM     100     A
CCGDBGM            RJ     =XRSR=                                               CCGDBGM     101     A
CCGDBGM            USE    DEBUG                                                CCGDBGM     102     A
CCGDBGM   X        CON    Y                                                    CCGDBGM     103     A
CCGDBGM            IFC    EQ,//BLK/,2                                          CCGDBGM     104     A
CCGDBGM            CON    0,=XL$TXT,0                                          CCGDBGM     105     A
CCGDBGM            ELSE   1                                                    CCGDBGM     106     A
CCGDBGM            CON    0,=XL$BLK,0                                          CCGDBGM     107     A
CCGDBGM   Y        SNAPCTRS (L BLK),LL,UL,INC                                  CCGDBGM     108     A
CCGDBGM            USE    *                                                    CCGDBGM     109     A
CCGDBGM   SNAPRL   ENDM                                                        CCGDBGM     110     A
CCGDBGM                                                                        CCGDBGM     111     A
CCGDBGM   SNAPCTRS MACRO  L,LL,UL,INC                                          CCGDBGM     112     A
CCGDBGM            CON    0,10H L                                              CCGDBGM     113     A
CCGDBGM            CON    LL 1                                                 CCGDBGM     114     A
CCGDBGM            CON    UL 200                                               CCGDBGM     115     A
CCGDBGM            CON    INC 1                                                CCGDBGM     116     A
CCGDBGM   SNAPCTRS ENDM                                                        CCGDBGM     117     A
CCGDBGM   DCALL    SPACE  3,24                                                 CCGDBGM     118     A
CCGDBGM  **        DCALL - CALL TO DEBUGGING ROUTINE CODED IN FTN OR SYMPL.    CCGDBGM     119     A
CCGDBGM  *                                                                     CCGDBGM     120     A
CCGDBGM  *         DCALL ROUTINE,(PARAMLIST)                                   CCGDBGM     121     A
CCGDBGM  *                                                                     CCGDBGM     122     A
CCGDBGM  *         [X] IS THE CONTENTS OF X.                                   CCGDBGM     123     A
CCGDBGM  *         CONSTANTS MUST BE WRITTEN AS =X  ( =3 ).                    CCGDBGM     124     A
CCGDBGM                                                                        CCGDBGM     125     A
CCGDBGM   DCALL    MACRO  RTN,P                                                CCGDBGM     126     A
CCGDBGM            LOCAL  APL                                                  CCGDBGM     127     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCGDBGM

```
CCGDBGM          RJ     =XSVR=                                          CCGDBGM    128    A
CCGDBGM          USE    DEBUG                                           CCGDBGM    129    A
CCGDBGM   APL    BSS    0                                               CCGDBGM    130    A
CCGDBGM          NOREF  .N                                              CCGDBGM    131    A
CCGDBGM   .N     SET    0                                               CCGDBGM    132    A
CCGDBGM  *                                                              CCGDBGM    133    A
CCGDBGM          IRP    P                                               CCGDBGM    134    A
CCGDBGM   .1     MICRO  1,1,/P/                                         CCGDBGM    135    A
CCGDBGM          IFC    EQ,/".1"/[/                                     CCGDBGM    136    A
CCGDBGM   .1     MICRO  2,,]_P_]                                        CCGDBGM    137    A
CCGDBGM          USE    *                                               CCGDBGM    138    A
CCGDBGM          SA2    =X".1"                                          CCGDBGM    139    A
CCGDBGM          SX6    X2+                                             CCGDBGM    140    A
CCGDBGM          SA6    APL+.N                                          CCGDBGM    141    A
CCGDBGM          USE    DEBUG                                           CCGDBGM    142    A
CCGDBGM          BSS    1                                               CCGDBGM    143    A
CCGDBGM          ELSE   4                                               CCGDBGM    144    A
CCGDBGM          IFC    EQ,/".1"/=/,2                                   CCGDBGM    145    A
CCGDBGM          VFD    42/,18/P                                        CCGDBGM    146    A
CCGDBGM          SKIP   1                                               CCGDBGM    147    A
CCGDBGM          VFD    42/,18/=X_P                                     CCGDBGM    148    A
CCGDBGM   .N     SET    .N+1                                            CCGDBGM    149    A
CCGDBGM          IRP                                                    CCGDBGM    150    A
CCGDBGM  *                                                              CCGDBGM    151    A
CCGDBGM          DATA   0                                               CCGDBGM    152    A
CCGDBGM          USE    *                                               CCGDBGM    153    A
CCGDBGM  *                                                              CCGDBGM    154    A
CCGDBGM          SA1    APL                                             CCGDBGM    155    A
CCGDBGM          RJ     =X_RTN                                          CCGDBGM    156    A
CCGDBGM          RJ     =XRSR=                                          CCGDBGM    157    A
CCGDBGM   DCALL  ENDM                                                   CCGDBGM    158    A
CCGDBGM   REGSNAP SPACE 3,14                                            CCGDBGM    159    A
CCGDBGM  **      REGSNAP - SNAP REGISTERS AT ENTRY POINTS.              CCGDBGM    160    A
CCGDBGM  *                                                              CCGDBGM    161    A
CCGDBGM  *       ACTIVATES REGISTER SNAP AT EACH ENTRY POINT DEFINED WITH THE  CCGDBGM    162    A
CCGDBGM  *       -ENTRY.- MACRO AND NAMED ON THE -REGSNAP- CALL CARD.   CCGDBGM    163    A
CCGDBGM  *       THE SNAP COUNT IS LIMITED TO 100D PER ENTRY POINT BY THE  CCGDBGM    164    A
CCGDBGM  *       -REGS- DEBUG MACRO.                                    CCGDBGM    165    A
CCGDBGM  *                                                              CCGDBGM    166    A
CCGDBGM  *                                                              CCGDBGM    167    A
CCGDBGM  *       REGSNAP   (NAME1,NAME2,...,NAMEN)                      CCGDBGM    168    A
CCGDBGM  *                                                              CCGDBGM    169    A
CCGDBGM  *       ENTRY  *NAMEK* = NAME OF ENTRY POINT TO BE SNAPPED     CCGDBGM    170    A
CCGDBGM  *                                                              CCGDBGM    171    A
CCGDBGM  *       EXIT   SNAP ACTIVATED                                  CCGDBGM    172    A
CCGDBGM  *                                                              CCGDBGM    173    A
CCGDBGM  *       USES   NONE                                            CCGDBGM    174    A
CCGDBGM  *                                                              CCGDBGM    175    A
CCGDBGM  *       CALLS  NONE                                            CCGDBGM    176    A
CCGDBGM                                                                 CCGDBGM    177    A
CCGDBGM                                                                 CCGDBGM    178    A
CCGDBGM          PURGMAC   REGSNAP                                      CCGDBGM    179    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCGDBGM

CCGDBGM                                                                  CCGDBGM      180     A
CCGDBGM   REGSNAP  MACRO  P                                              CCGDBGM      181     A
CCGDBGM            IRP    P                                              CCGDBGM      182     A
CCGDBGM   )P       =      1           DEFINE SYMBOL                     CCGDBGM      183     A
CCGDBGM            IRP                                                   CCGDBGM      184     A
CCGDBGM   REGSNAP  ENDM                                                  CCGDBGM      185     A
CCGDBGM   SNAPT    SPACE  3,12                                           CCGDBGM      186     A
CCGDBGM  **        SNAPT - SNAP A TABLE WITH POINTERS O.TBL AND L.TBL.   CCGDBGM      187     A
CCGDBGM                                                                  CCGDBGM      188     A
CCGDBGM            PURGMAC   SNAPT                                       CCGDBGM      189     A
CCGDBGM                                                                  CCGDBGM      190     A
CCGDBGM   SNAPT    MACRO  TBL,BCD                                        CCGDBGM      191     A
CCGDBGM   BCD      SNAP   *O$TBL,,*L$TBL,NOREGS                          CCGDBGM      192     A
CCGDBGM   SNAPT    ENDM                                                  CCGDBGM      193     A
CCGDBGM                                                                  CCGDBGM      194     A
CCGDBGM            ENDX                                                  CCGDBGM      195     A

          SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - CCGDBGM

                IDENTIFIER                TOTAL    ACTIVE

                CCGDBGM                     195       195

                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCGILFD

CCGILFD  *COMDECK  CCGILFD            CCG *IL* FIELD DEFINITIONS         CCGILFD        1     A
CCGILFD   DESCR    TITLE  CCG *IL* FIELD DEFINITIONS AND TABLE FORMATS   CCGILFD        2     A
CCGILFD  **        *IL* INSTRUCTION DESCRIPTOR FIELDS ( ALL INSTRUCTION TYPES )  CCGILFD  3     A
CCGILFD  *         INSTRUCTION ATTRIBUTES ( PROPERTY BITS ).             CCGILFD        4     A
CCGILFD                                                                  CCGILFD        5     A
CCGILFD            DESCRIBE D.,60                                        CCGILFD        6     A
CCGILFD   TY       DEFINE 2            RLIST TYPE - 1 ( = 0-3 )          CCGILFD        7     A
CCGILFD   FT       DEFINE 5            FUNCTIONAL UNIT TIME IN MINOR CYCLES  CCGILFD    8     A
CCGILFD   NR       DEFINE 1            D.RJRSP+12 ( BIT SHIFT FOR *MCG* IN *DRR* )  CCGILFD  9  A
CCGILFD            DEFINE 1            =1 TO SIMULATE A EXPONENT         CCGILFD       10     A
CCGILFD   F1       DEFINE 4            FIRST FUNCTION UNIT ORDINAL       CCGILFD       11     A
CCGILFD   F2       DEFINE 4            SECOND FUNCTION UNIT ORDINAL      CCGILFD       12     A
CCGILFD   BM       DEFINE 1            NODE BOUNDARY MARKER ( JUMP, LABEL, ETC )  CCGILFD  13  A
CCGILFD   ^D       DEFINE 1            SET IF INSTRUCTION DOESN-T DEF AN RI/ NO RI  CCGILFD  14  A
CCGILFD                                                                  CCGILFD       15     A
CCGILFD  **        NOTE - THE BIT FIELD FROM *RJRS* TO *JFT* IS USED BY THE  CCGILFD  16     A
CCGILFD  *         SCHEDULER TO ASSIGN RESULT REGISTERS, SO PLEASE DO NOT REORDER  CCGILFD  17  A
CCGILFD                                                                  CCGILFD       18     A
CCGILFD   RJRS     DEFINE 1            PRECEDES AN RJ REGISTER STORE     CCGILFD       19     A
CCGILFD   PS       DEFINE 1            INST IS AN IMMEDIATE PREDECESSOR OF A STORE  CCGILFD  20  A
CCGILFD   PRS      DEFINE 1            PRECEDES A REGISTER STORE         CCGILFD       21     A
CCGILFD   ECJ      DEFINE 1            ERROR CHECK (FALL THROUGH) JUMP   CCGILFD       22     A
CCGILFD   L2       DEFINE 1            LEVEL 2 LD/ST                     CCGILFD       23     A
CCGILFD   LD       DEFINE 1            LOAD                              CCGILFD       24     A
CCGILFD   ST       DEFINE 1            STORE                             CCGILFD       25     A
CCGILFD   JP       DEFINE 1            JUMP                              CCGILFD       26     A
CCGILFD   XMT      DEFINE 1            XMT/SXT                           CCGILFD       27          I

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCGILFD

-CCG0041

```
CCGILFD   XMT      DEFINE 1          XMT/SXT (LEVEL 0 LD/ST IF *L2* SET)              CCG0041      9     A
CCGILFD   KS       DEFINE 1          OPERATION DESTROYS OPERAND ( CON SHIFTS )        CCGILFD     28     A
CCGILFD   PI       DEFINE 1          PSEUDO INSTRUCTION ( RS,DAR,DEF,LAB, ETC )       CCGILFD     29     A
CCGILFD   MUC      DEFINE 1          MULTI USE COMPUTATION ( *JAM* MODE )             CCGILFD     30     A
CCGILFD   JFT      DEFINE 1          JAM FUNCTION TIME, "0 IF USES A FUNCTION UN      CCGILFD     31     A
CCGILFD                                                                               CCGILFD     32     A
CCGILFD   USI      DEFINE 1          UNSAFE INSTRUCTION ( INTERRUPT POSSIBLE )        CCGILFD     33     A
CCGILFD   ZP       DEFINE 1          =1 IF INSTRUCTION HAS NO OPERANDS                CCGILFD     34     A
CCGILFD   CM       DEFINE 1          "0 IF OPERANDS ARE COMMUTATIVE                   CCGILFD     35     A
CCGILFD   SZ       DEFINE 2          INSTRUCTION SIZE ( PARCEL COUNT )                CCGILFD     36     A
CCGILFD   SR       DEFINE 1          SET FOR SIZE REDUCIBLE LD/ST-S                   CCGILFD     37     A
CCGILFD   FP       DEFINE 1          SET IF IH IS AN F.P.                             CCGILFD     38     A
CCGILFD   XU       DEFINE 1          USES EXTEND PAST A REDEF ( SIO AND GRA )         CCGILFD     39     A
CCGILFD   RF       DEFINE 1          RI USED IN *RF* OF LD/ST/STT/SA                  CCGILFD     40     A
CCGILFD   INC      DEFINE 1          INCREMENT INSTRUCTION ( IA/IS/STT/ST )           CCGILFD     41     A
CCGILFD   REGW     DEFINE 9          X-REGISTER WIDTH OF THE SEQUENCE AT A INST       CCGILFD     42     A
CCGILFD   USES     DEFINE 9          USES COUNT                                       CCGILFD     43     A
CCGILFD                                                                               CCGILFD     44     A
CCGILFD   *        COMBINATIONS OF DESCRIPTOR BIT FIELDS                               CCGILFD     45     A
CCGILFD                                                                               CCGILFD     46     A
CCGILFD   LDST     DEQU   ST,2       LD AND ST BITS ( MEMORY REFERENCE )              CCGILFD     47     A
CCGILFD   STRS     DEQU   PRS,2      STORE / RS PRECEDENCE BITS                       CCGILFD     48     A
CCGILFD   AR       DEQU   FP,2       ADDRESS REDUCTION BITS ( *MCG* )                 CCGILFD     49     A
CCGILFD   LSJ      DEQU   JP,3       LD / ST / JP BITS                                CCGILFD     50     A
CCGILFD   TYIO     DEQU   RJRS,2     TYPE I OPERAND BIT DETERMINOR                    CCGILFD     51     A
CCGILFD   *                          0 - RJ,RK ARE OPERANDS                           CCGILFD     52     A
CCGILFD   *                          1 - RK IS AN OPERAND                             CCGILFD     53     A
CCGILFD   *                          2 - RI,RJ,RK ARE OPERANDS                        CCGILFD     54     A
CCGILFD   EQV      DEQU   FT,1       "0 IF INST IS PRED OF AN EQUIV LINK              CCGILFD     55     A
CCGILFD   *                ASSUMES THAT FUN TIME OF LD/ST-S IS AN EVEN NUMBER.        CCGILFD     56     A
CCGILFD   RLIST    EJECT                                                              CCGILFD     57     A
CCGILFD   **       *IL* INSTRUCTION FIELD DEFINITIONS                                 CCGILFD     58     A
CCGILFD                                                                               CCGILFD     59     A
CCGILFD            DESCRIBE R1.,60   TYI(OC,RJ,RK,RI) / TYII(OC,IN,SO,RI) , ETC       CCGILFD     60     A
CCGILFD                                                                               CCGILFD     61     A
CCGILFD   *        TYPE 1 *IL* - BINARY OPERATIONS - RI = RJ.OP.RK                    CCGILFD     62     A
CCGILFD   *                12/PACKED OC,16/RJ,16/RK,16/RI                             CCGILFD     63     A
CCGILFD                                                                               CCGILFD     64     A
CCGILFD   OC       DEFINE 12         PACKED OP CODE                                   CCGILFD     65     A
CCGILFD   RJ       DEFINE 16         FIRST OPERAND                                    CCGILFD     66     A
CCGILFD   RK       DEFINE 16         SECOND OPERAND                                   CCGILFD     67     A
CCGILFD   RI       DEFINE 16         RESULT RNUMBER                                   CCGILFD     68     A
CCGILFD                                                                               CCGILFD     69     A
CCGILFD   *        TYPE 2 *IL* - 1 ADDRESS AND PSEUDOS - .OP.RI = IN                  CCGILFD     70     A
CCGILFD   *                12/PACKED OC,18/IN,14/SO,16/RI                             CCGILFD     71     A
CCGILFD                                                                               CCGILFD     72     A
CCGILFD   IN       DEFINE 18,R1.RJP+15      A CA FIELD                                CCGILFD     73     A
CCGILFD   SO       DEFINE 14         REGISTER DESIGNATOR                              CCGILFD     74     A
CCGILFD                                                                               CCGILFD     75     A
CCGILFD   *        TYPE 3 *IL* - LOADS,STORES AND JUMPS                               CCGILFD     76     A
CCGILFD   *                12/PACKED OC,18/IN,12/H2,2/0,16/RI                         CCGILFD     77     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCGILFD

```
CCGILFD  *                R2 WORD = IHINFO FORMAT ( SEE BELOW )                CCGILFD      78    A
CCGILFD            SPACE  2                                                      CCGILFD      79    A
CCGILFD  *         TYPE 4 *IL* - SPECIAL INSTRUCTIONS, BOUNDARY MARKERS, ETC.    CCGILFD      80    A
CCGILFD  *                12/PACKED OC,18/CA,12/H2,18/IH                         CCGILFD      81    A
CCGILFD                                                                          CCGILFD      82    A
CCGILFD            DESCRIBE R1.,60    TYIV(OC,CA,H2,IH)                          CCGILFD      83    A
CCGILFD   OC       DEFINE 12                                                     CCGILFD      84    A
CCGILFD   CA       DEFINE 18                                                     CCGILFD      85    A
CCGILFD   H2       DEFINE 12          SYMTAB ORDINAL, MISCELLANEOUS INFO         CCGILFD      86    A
CCGILFD   IH       DEFINE 18          IH FIELD                                   CCGILFD      87    A
CCGILFD   IHINFO   SPACE  3                                                      CCGILFD      88    A
CCGILFD  **        IH INFO WORD/ R2 WORD FOR MEMORY REFERENCES AND TYPE III INST CCGILFD      89    A
CCGILFD  *         NOTE - LD,ST,BM BITS SET BY *BDT* IN *FIL* DURING INTERFERENCE CCGILFD      90    A
CCGILFD  *                LINKING PHASE ONLY ( BITS ARE SET IN *IH* TABLE ).     CCGILFD      91    A
CCGILFD  *         LD BIT SET FOR APLIST LD-S IN *PRE* TO PREVENT SQUEEZING OF   CCGILFD      92    A
CCGILFD  *                ST/LD SEQUENCES IN *SQZ*.                              CCGILFD      93    A
CCGILFD                                                                          CCGILFD      94    A
CCGILFD            DESCRIBE IH.,60    IHW(RF,CA,IH)                              CCGILFD      95    A
CCGILFD   LD       DEFINE 1           LD BIT FROM DESCR                          CCGILFD      96    A
CCGILFD   ST       DEFINE 1           ST BIT FROM DESCR                          CCGILFD      97    A
CCGILFD   BM       DEFINE 1           SET FOR BOUNDARY MARKERS                   CCGILFD      98    A
CCGILFD            DEFINE 2                                                      CCGILFD      99          I
          -FCC1918
CCGILFD            DEFINE 1                                                      FCC1918       7    A
CCGILFD   SIA      DEFINE 1           STT GEN BY *GRA* OF LCM&FP  ADDRESS        FCC1918       8    A
CCGILFD  *                              WHERE VALUE OF RF MAY BE NEGATIVE        FCC1918       9    A
CCGILFD   SRF      DEFINE 1           =1 IF SPECIAL RF NECESSARY ( RF = (RJ,RK) ) CCGILFD     100    A
CCGILFD   RF       DEFINE 18          R-NUMBER OF MODIFIER                       CCGILFD     101    A
CCGILFD   CA       DEFINE 18          CONSTANT ADDEND                            CCGILFD     102    A
CCGILFD   I        DEFINE 3           TABLE NUMBER ( 0 - SYMTAB, ETC )           CCGILFD     103    A
CCGILFD   H        DEFINE 15          ORDINAL INTO TABLE                         CCGILFD     104    A
CCGILFD   IH       DEQU   H,18                                                   CCGILFD     105    A
CCGILFD                                                                          CCGILFD     106    A
CCGILFD   CAIH     DEQU   H,36        COMBINATION OF CA AND IH FIELDS            CCGILFD     107    A
CCGILFD                                                                          CCGILFD     108    A
CCGILFD   IH.LCMA  EQU    1S13        =1 IF LCM=I ADDRESS LOAD ( ^F.P. )         CCGILFD     109    A
CCGILFD   IH.LCM   EQU    1S14        *LCM* ( LEVEL 2 ) BIT VALUE SET IN *IH*    CCGILFD     110    A
CCGILFD  *                            FIELD FOR *DRL/DWL*-S .                    CCGILFD     111    A
CCGILFD   RF.MV    EQU    1S16        BIT SET IN *RF* FIELD OF PLD,PST,DRL,DWL   CCGILFD     112    A
CCGILFD  *                            IF RF IS -VAR  ( A(C-VAR)  ).              CCGILFD     113    A
CCGILFD   R2.LONGP EQU    58D         BIT =1 IN R2 OF *BOS* IF FIRST BLOCK OF A  CCG21         1    A
CCGILFD  *                            DO LOOP AND LOOP INDEX IS *LONG*           CCG21         2    A
CCGILFD   IH.STKP  EQU    59          =1 IN R2 OF LOOP GEN LABEL IF LOOP WILL    CCG29         1    A
CCGILFD  *                              FIT INSTACK ( GRA AND MCG )              CCG29         2    A
CCGILFD   IFIELDS  SPACE  2,6                                                    CCGILFD     114    A
CCGILFD   I.GL     EQU    1S15        I FIELD VALUE OF GL.NNN SYMBOLS            CCGILFD     115    A
CCGILFD   RN.MASK  EQU    600001B     MASK FOR EXTRACTING R-NUMBERS              CCGILFD     116    A
CCGILFD   SOREG    SPACE  3                                                      CCGILFD     117    A
CCGILFD  **        SO FIELD  - USED FOR SPECIFYING EXPLICIT RESULT REGISTERS    CCGILFD     118    A
CCGILFD  *                     IN *RS* AND *DEF* INSTRUCTIONS ONLY.              CCGILFD     119    A
CCGILFD  *                2/INV INFO,2/LOCK TYPE,3/REG TYPE,3/REG NUMBER         CCGILFD     120    A
CCGILFD                                                                          CCGILFD     121    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCGILFD

```
CCGILFD            DESCRIBE SO.,11                                          CCGILFD    122    A
CCGILFD   SFL      DEFINE 1          SCRATCHABLE FULL LOCK ( TEMP/FULL LOCK )  CCGILFD    123    A
CCGILFD   INVC     DEFINE 1          INVARIANT IN CURRENT LOOP                 CCGILFD    124    A
CCGILFD   INV      DEFINE 1          INVARIANT IN INNER LOOP                   CCGILFD    125    A
CCGILFD   LK       DEFINE 2          LOCK TYPE, *RS* INSTRUCTION ONLY          CCGILFD    126    A
CCGILFD  *                           0 - HOLD UNTIL NEXT UNCONDITIONAL JUMP    CCGILFD    127    A
CCGILFD  *                           1 - HOLD UNTIL USES[RI] = 0               CCGILFD    128    A
CCGILFD  *                           2 - HOLD UNTIL REDEF OR *EOQ*             CCGILFD    129    A
CCGILFD  *                           3 - RJ RS, IT IS A *DEF* IN DISGUISE.     CCGILFD    130    A
CCGILFD  *                           3 - A-REG LOCK ( INITIAL *DEF* OF PREFETCH)  CCGILFD    131    A
CCGILFD   RT       DEFINE 3          REGISTER TYPE ( BAX = 0/1/2 )             CCGILFD    132    A
CCGILFD   RN       DEFINE 3          REGISTER NUMBER ( 0 - 7 )                 CCGILFD    133    A
CCGILFD   REG      DEQU   RN,6       RT + RN FIELDS                            CCGILFD    134    A
CCGILFD                                                                       CCGILFD    135    A
CCGILFD   .LKP     DECMIC SO.LKP                                               CCGILFD    136    A
CCGILFD   .LKP     SET    1S".LKP"                                             CCGILFD    137    A
CCGILFD   SO.TLOCK EQU    .LKP       TEMP LOCK                                 CCGILFD    138    A
CCGILFD  *                             WARNING - TEMP LOCKS CAN BE ELIMINATED  CCGILFD    139    A
CCGILFD  *                             IN *SQZ* IF THEY HAVE NO *REAL* USES.   CCGILFD    140    A
CCGILFD   SO.LOCK  EQU    2*.LKP     FULL LOCK ( UNTIL REDEF / *EOQ* )         CCGILFD    141    A
CCGILFD   SO.RJLK  EQU    3*.LKP     RJ LOCK ( B-REG DEF FOR UP / NR INSTRUCTION  CCGILFD    142    A
CCGILFD                                                                       CCGILFD    143    A
CCGILFD   RT.BR    EQU    0          VALUE OF B REGISTER TYPE                  CCGILFD    144    A
CCGILFD   RT.XR    EQU    2          VALUE OF X REGISTER TYPE                  CCGILFD    145    A
CCGILFD   APL      SPACE  3,10                                                  CCGILFD    146    A
CCGILFD  **        AP / IO LIST ENTRY FORMAT FOR USE/DEF INFO PROCESSING       CCGILFD    147    A
CCGILFD                                                                       CCGILFD    148    A
CCGILFD            DESCRIBE AP.,60                                            CCGILFD    149    A
CCGILFD   IO       DEFINE 1          =1 IF INPUT ( DEF ) IN *IOL*              CCGILFD    150    A
CCGILFD   USE      DEFINE 1          =1 IF INPUT LIST ITEM MAY NOT BE DEFINED  CCGILFD    151    A
CCGILFD   P1       DEFINE 1          +1 REF FOR A DOUBLE/COMPLEX VARIABLE      CCGILFD    152    A
CCGILFD            DEFINE 2                                                   CCGILFD    153    A
CCGILFD   CR       DEFINE 1          CLASS REFERENCE                           CCGILFD    154    A
CCGILFD            DEFINE 18                                                  CCGILFD    155    A
CCGILFD   CA       DEFINE 18         BIAS                                      CCGILFD    156    A
CCGILFD   IH       DEFINE 18         SYMBOL ORDINAL                            CCGILFD    157    A
CCGILFD                                                                       CCGILFD    158    A
CCGILFD   CAIH     DEQU   IH,36                                               CCGILFD    159    A
CCGILFD   FI       SPACE  3,14                                                CCGILFD    160    A
CCGILFD  **        FI. - FUNCTION INFORMATION WORD ( R2 ) FOR RJX-S.          CCGILFD    161    A
CCGILFD                                                                       CCGILFD    162    A
CCGILFD            DESCRIBE FI.,60                                            CCGILFD    163    A
CCGILFD   FT       DEFINE 6          FUNCTION TYPE                             CCGILFD    164    A
CCGILFD  *                           0 - IGNORE, 1 - USER, 2 - BEF, 3 - I/O    CCGILFD    165    A
CCGILFD   REGP     DEFINE 24         BIT MASK OF REGISTERS PRESERVED           CCGILFD    166    A
CCGILFD  *                           ( 0 - B0 , 1 - B1, ... 23 - X7 )          CCGILFD    167    A
CCGILFD   INDX     DEFINE 18         INDEX TO LIST IN *IOL*                    CCGILFD    168    A
CCGILFD   LEN      DEFINE 12         LENGTH OF LIST IN *IOL* + 1S10 IF " 0     CCGILFD    169    A
CCGILFD   SI       SPACE  2,14                                                CCGILFD    170    A
CCGILFD  **        SI. - *SLIST* FILE FORMAT FOR MACHINE INSTRUCTIONS         CCGILFD    171    A
CCGILFD                                                                       CCGILFD    172    A
CCGILFD            DESCRIBE SI.                                               CCGILFD    173    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCGILFD

```
CCGILFD   H2       DEFINE 1           =1 IF WORD WITH *H2* FOLLOWS THIS WORD          CCGILFD      174     A
CCGILFD   OPC      DEFINE 11          PACKED OPCODE                                   CCGILFD      175     A
CCGILFD   CA       DEFINE 18                                                          CCGILFD      176     A
CCGILFD   IH       DEFINE 18                                                          CCGILFD      177     A
CCGILFD   RJ       DEFINE 6           J REGISTER NUMBER IN SO. FORMAT                 CCGILFD      178     A
CCGILFD   RI       DEFINE 6           I REGISTER NUMBER IN SO. FORMAT                 CCGILFD      179     A
CCGILFD                                                                               CCGILFD      180     A
CCGILFD   RK       DEQU   IH,6        K REGISTER NUMBER IN SO. FORMAT                 CCGILFD      181     A
CCGILFD   CAIH     DEQU   IH,36                                                       CCGILFD      182     A
CCGILFD   OPR      SPACE  2,14                                                        CCGILFD      183     A
CCGILFD  **        OPR - MACRO TO DEFINE *IL* INSTRUCTION OPCODES ( OC. SYMBOLS )    CCGILFD      184     A
CCGILFD                                                                               CCGILFD      185     A
CCGILFD            MACRO  OPR,N                                                       CCGILFD      186     A
CCGILFD   OC.N     EQU    .OC                                                         CCGILFD      187     A
CCGILFD   .OC      SET    .OC+1                                                       CCGILFD      188     A
CCGILFD   OPR      ENDM                                                               CCGILFD      189     A
CCGILFD                                                                               CCGILFD      190     A
CCGILFD            NOREF  .OC                                                         CCGILFD      191     A
CCGILFD   .OC      SET    0                                                           CCGILFD      192     A
CCGILFD                                                                               CCGILFD      193     A
CCGILFD            SKIP   1           SO SYMBOLS ARE PUT IN STEXT RECORD              CCGILFD      194     A
CCGILFD  *CALL     OPRDEFS                                                            CCGILFD      195     A
CCGILFD                                                                               CCGILFD      196     A
CCGILFD  *         DEFINE SUBINDEX OPCODES FOR THE X AND B JUMPS ( IN *CA* FIELD     CCGILFD      197     A
CCGILFD                                                                               CCGILFD      198     A
CCGILFD            NOREF  .Z                                                          CCGILFD      199     A
CCGILFD   .Z       SET    0                                                           CCGILFD      200     A
CCGILFD            ECHO   2,T=("XJUMPS")                                              CCGILFD      201     A
CCGILFD   JC.T     EQU    .Z                                                          CCGILFD      202     A
CCGILFD   .Z       SET    .Z+1                                                        CCGILFD      203     A
CCGILFD                                                                               CCGILFD      204     A
CCGILFD   .Z       SET    0                                                           CCGILFD      205     A
CCGILFD            ECHO   2,T=("BJUMPS")                                              CCGILFD      206     A
CCGILFD   JC.T     EQU    .Z                                                          CCGILFD      207     A
CCGILFD   .Z       SET    .Z+1                                                        CCGILFD      208     A
CCGILFD   PODEFS   SPACE  3,8                                                         CCGILFD      209     A
CCGILFD  **        POD - MACRO TO DEFINE THE *SLIST* INSTRUCTION OPCDOES             CCGILFD      210     A
CCGILFD                                                                               CCGILFD      211     A
CCGILFD            MACRO  POD,NAM                                                     CCGILFD      212     A
CCGILFD   OC.NAM   EQU    .OC                                                         CCGILFD      213     A
CCGILFD   .OC      SET    .OC-1                                                       CCGILFD      214     A
CCGILFD            ENDM                                                               CCGILFD      215     A
CCGILFD                                                                               CCGILFD      216     A
CCGILFD   .OC      SET    -1                                                          CCGILFD      217     A
CCGILFD            SKIP   1                                                           CCGILFD      218     A
CCGILFD  *CALL     PSODEFS                                                            CCGILFD      219     A
CCGILFD   CF.      SPACE  3,14                                                        CCG16          2     A
CCGILFD                                                                               CCG16          3     A
CCGILFD            DESCRIBE CF.,60                                                    CCG16          4     A
CCGILFD   JP       DEFINE 1                                                           CCG16          5     A
CCGILFD            DEFINE 11                                                          CCG16          6     A
CCGILFD   FROM     DEFINE 18          BLOCK NUMBER OF PRED                            CCG16          7     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCGILFD

CCGILFD            DEFINE 12                                                CCG16          8     A
CCGILFD   TO       DEFINE 18          BLOCK NUMBER OF SUCC                  CCG16          9     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - CCGILFD

          IDENTIFIER                TOTAL    ACTIVE

          CCGILFD                     219       217
          CCG16                         8         8
          FCC1918                       3         3
          CCG21                         2         2
          CCG29                         2         2
          CCG0041                       1         1

          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SMOCDEF

SMOCDEF  *COMDECK  SMOCDEF                                                  SMOCDEF        1     A
SMOCDEF   SMOCDEF  CTEXT  SMOCDEF - SMACRO OPCODE VALUE DEFINITION MACROS   SMOCDEF        2     A
SMOCDEF  **        SMOCDEF - DECK TO BE CALLED PRIOR TO CALLING *SMACROS* TO SMOCDEF       3     A
SMOCDEF  *         ESTABLISH THE OPCODE VALUES OF THE HOST DEFINED *SMACROS*. SMOCDEF      4     A
SMOCDEF  *         FOR A MACRO NAME *N* IT DEFINES THE OPCDOE VALUE OF THE   SMOCDEF       5     A
SMOCDEF  *         NAME AS M$N .                                             SMOCDEF       6     A
SMOCDEF                                                                     SMOCDEF       7     A
SMOCDEF            MACRO  SMACRO,N                                          SMOCDEF       8     A
SMOCDEF   M$N      =      .Z                                                SMOCDEF       9     A
SMOCDEF   .Z       SET    .Z+1                                              SMOCDEF      10     A
SMOCDEF            ENDM                                                     SMOCDEF      11     A
SMOCDEF                                                                     SMOCDEF      12     A
SMOCDEF            NOREF  .Z                                                SMOCDEF      13     A
SMOCDEF   .Z       SET    128                                               SMOCDEF      14     A
SMOCDEF                                                                     SMOCDEF      15     A
SMOCDEF            ECHO   1,Z=(BTW,ETW,ARI,ACI,MIC,ASV,ELSE.,ENDIF.,SET.,DCSS,___ SMOCDEF  16     A
SMOCDEF  ,USE.,QUAL.,CON.,BSS.,BSSZ.,ENDS)                                  SMOCDEF      17     A
SMOCDEF   Z        OPSYN  NIL                                               SMOCDEF      18     A
SMOCDEF            ECHO   1,Z=(LT,EQ,GT,LE,NE,GE)                           SMOCDEF      19     A
SMOCDEF   IF.Z     OPSYN  NIL                                               SMOCDEF      20     A
SMOCDEF                                                                     SMOCDEF      21     A
SMOCDEF            ENDX                                                     SMOCDEF      22     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - SMOCDEF

          IDENTIFIER                TOTAL    ACTIVE

          SMOCDEF                      22        22

          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SSTCALL

SSTCALL  *COMDECK  SSTCALL                                                  SSTCALL        1     A
SSTCALL                                                                     SSTCALL        2     A
SSTCALL            CCG.SST                                                  SSTCALL        3     A
SSTCALL            LIST   F                                                 SSTCALL        4     A
SSTCALL                                                                     SSTCALL        5     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - SSTCALL

         IDENTIFIER                TOTAL    ACTIVE

         SSTCALL                       5         5

         LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN *TEXTS*

*TEXTS*  *DECK     *TEXTS*                                                  *TEXTS*        1     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - *TEXTS*

         IDENTIFIER                TOTAL    ACTIVE

         *TEXTS*                       1         1

         LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CMPLTXT

CMPLTXT  *DECK     CMPLTXT                                                  CMPLTXT        1     A
CMPLTXT            IDENT  CMPLTXT                                           CMPLTXT        2     A
CMPLTXT   CMPLTXT  TITLE  CMPLTXT - COMPILER PRODUCT"S ASSEMBLY TEXT        CMPLTXT        3     A
CMPLTXT            STEXT                                                    CMPLTXT        4     A
CMPLTXT            SST                                                      CMPLTXT        5     A
CMPLTXT            LIST   F,X                                               CMPLTXT        6     A
CMPLTXT            COMMENT  COMPILER PRODUCT"S ASSEMBLY TEXT   190=   SPAC  CMPLTXT        7     A
CMPLTXT            SPACE  3                                                 CMPLTXT        8     A
CMPLTXT  ***       CMPLTXT - COMPILER PRODUCT"S ASSEMBLY TEXT FOR COMPILERS CMPLTXT        9     A
CMPLTXT  *         THAT USE CCG ( COMMON CODE GENERATOR ).                  CMPLTXT       10     A
CMPLTXT  *                                                                  CMPLTXT       11     A
CMPLTXT  *         THIS TEXT CONTAINS SYMBOL, MACRO AND MICRO DEFINITIONS   CMPLTXT       12     A
CMPLTXT  *         NECESSARY TO ASSEMBLE THE HOST COMPILER.                 CMPLTXT       13     A
CMPLTXT  *         THE TEXT IS DIVIDED UP INTO THE FOLLOWING SECTIONS -     CMPLTXT       14     A
CMPLTXT  *          INSTALLATION DEPENDENT INSTALLATION PARAMETERS.         CMPLTXT       15     A
CMPLTXT  *          STRUCTURED FIELD DEFINITION MACROS.                     CMPLTXT       16     A
CMPLTXT  *          I/O CODING CONVENTIONS AND MACROS                       CMPLTXT       17     A
CMPLTXT  *          GENERAL MACROS                                          CMPLTXT       18     A
CMPLTXT  *          MACROS FOR DEBUGGING                                    CMPLTXT       19     A
CMPLTXT  *         CCG FIELD DEFINITIONS, EQU"S AND MACROS.                 CMPLTXT       20     A
CMPLTXT  *CALL     OPTIONS                                                  CMPLTXT       21     A
CMPLTXT  *CALL     COMADEF                                                  CMPLTXT       22     A
CMPLTXT  *CALL     CCOMGCM                                                  CMPLTXT       23     A
CMPLTXT  *CALL     CCGILFD                                                  CMPLTXT       24     A
CMPLTXT   WC       SPACE  3,14                                              CMPLTXT       25     A
CMPLTXT  **        WC. - FORMAT OF WORD C OF SYMTAB FOR ADDRESS DEFINITION  CMPLTXT       26          I
         -CCG16
CMPLTXT                                                                     CMPLTXT       27          I
         -CCG16
CMPLTXT            DESCRIBE WC.                                             CMPLTXT       28          I
         -CCG16
CMPLTXT   RL       DEFINE 2          0 - ABS, 1 - PROG, 2 - COM, 3 - EXT    CMPLTXT       29          I
         -CCG16
CMPLTXT            DEFINE 13                                                CMPLTXT       30          I
         -CCG16
CMPLTXT   QBN      DEFINE 12         QUAL BLOCK NUMBER                      CMPLTXT       31          I

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CMPLTXT

          -CCG16
CMPLTXT    RB       DEFINE 9          BLOCK NUMBER ( COMPILER VALUE )                CMPLTXT      32          I
          -CCG16
CMPLTXT    RA       DEFINE 24         BLOCK/PROGRAM RELATIVE ADDRESS                 CMPLTXT      33          I
          -CCG16
CMPLTXT    WRITEP   SPACE  3,14                                                       CMPLTXT      34     A
CMPLTXT   **        WRITEP - WRITE PSEUDO OP WORD TO *SLIST* FILE                   CMPLTXT      35     A
CMPLTXT   *                                                                          CMPLTXT      36     A
CMPLTXT   *         ENTRY  (X6) = LOW 48 BITS OF PSEUDO OP WORD                      CMPLTXT      37     A
CMPLTXT   *                                                                          CMPLTXT      38     A
CMPLTXT   *         WRITEP OPNAME                                                    CMPLTXT      39     A
CMPLTXT                                                                              CMPLTXT      40     A
CMPLTXT    WRITEP   MACRO  OPCD                                                      CMPLTXT      41     A
CMPLTXT             R=     B3,OC.OPCD                                                CMPLTXT      42     A
CMPLTXT             RJ     =XWPW#                                                    CMPLTXT      43     A
CMPLTXT             ENDM                                                             CMPLTXT      44     A
CMPLTXT    ADDWRD   SPACE  3                                                         CMPLTXT      45     A
CMPLTXT   **        ADDWRD - ADD WORD TO MANAGED TABLE.                              CMPLTXT      46     A
CMPLTXT   *                                                                          CMPLTXT      47     A
CMPLTXT   *         ADDWRD TABLE,REG                                                 CMPLTXT      48     A
CMPLTXT   *         ENTRY  *TABLE* = TABLE NUMBER.                                   CMPLTXT      49     A
CMPLTXT   *                *REG* = REGISTER NAME OR EXPRESSION FOR WORD TO ADD.     CMPLTXT      50     A
CMPLTXT   *         USES   A0, X1.                                                   CMPLTXT      51     A
CMPLTXT                                                                              CMPLTXT      52     A
CMPLTXT             PURGMAC ADDWRD                                                   CMPLTXT      53     A
CMPLTXT    ADDWRD   MACRO  A,B                                                       CMPLTXT      54     A
CMPLTXT             R=     X1,B                                                      CMPLTXT      55     A
CMPLTXT             R=     A0,=XZ$A                                                  CMPLTXT      56     A
CMPLTXT             RJ     =XADW$                                                    CMPLTXT      57     A
CMPLTXT    ADDWRD   ENDM                                                             CMPLTXT      58     A
CMPLTXT    ALLOC    SPACE  3                                                         CMPLTXT      59     A
CMPLTXT   **        ALLOC - ALLOCATE TABLE SPACE.                                    CMPLTXT      60     A
CMPLTXT   *                                                                          CMPLTXT      61     A
CMPLTXT   *         ALLOC  TABLE,WORDS                                               CMPLTXT      62     A
CMPLTXT   *         ENTRY  *TABLE* = TABLE NUMBER.                                   CMPLTXT      63     A
CMPLTXT   *                *WORDS* = WORD COUNT OF TABLE.                           CMPLTXT      64     A
CMPLTXT                                                                              CMPLTXT      65     A
CMPLTXT             PURGMAC ALLOC                                                    CMPLTXT      66     A
CMPLTXT    ALLOC    MACRO  A,B                                                       CMPLTXT      67     A
CMPLTXT             R=     X1,B                                                      CMPLTXT      68     A
CMPLTXT             R=     A0,=XZ$A                                                  CMPLTXT      69     A
CMPLTXT             RJ     =XATS$                                                    CMPLTXT      70     A
CMPLTXT    ALLOC    ENDM                                                             CMPLTXT      71     A
CMPLTXT                                                                              CMPLTXT      72     A
CMPLTXT             END                                                              CMPLTXT      73     A
CMPLTXT   *WEOR                                                                      CMPLTXT      74          I
          -CCGA054

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - CMPLTXT

          IDENTIFIER                TOTAL    ACTIVE

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
            LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CMPLTXT

            CMPLTXT                    74        65


            LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN WEOR1*


WEOR1*      *DECK WEOR1*                                                    CCGA054        5     A
WEOR1*      *WEOR 0                                                         CCGA054        6     A


        SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - WEOR1*

            IDENTIFIER                TOTAL    ACTIVE

            CCGA054                     2         2


            LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCGTEXT

CCGTEXT     *DECK     CCGTEXT                                               CCGTEXT        1     A
CCGTEXT               IDENT  CCGTEXT                                        CCGTEXT        2     A
CCGTEXT               TITLE  CCGTEXT - COMMON CODE GENERATOR ASSEMBLY TEXT  CCGTEXT        3     A
CCGTEXT               STEXT                                                 CCGTEXT        4     A
CCGTEXT               SST                                                   CCGTEXT        5     A
CCGTEXT               LIST   F,X                                            CCGTEXT        6     A
CCGTEXT               COMMENT  COMMON CODE GENERATOR ASSEMBLY TEXT          CCGTEXT        7     A
CCGTEXT               SPACE  3                                              CCGTEXT        8     A
CCGTEXT     ***       CCGTEXT - COMMON CODE GENERATOR ( CCG ) ASSEMBLY TEXT CCGTEXT        9     A
CCGTEXT     *                                                               CCGTEXT       10     A
CCGTEXT     *         THIS TEXT CONTAINS SYMBOL, MACRO AND MICRO DEFINITIONS CCGTEXT       11     A
CCGTEXT     *         NECESSARY TO ASSEMBLE CCG.                            CCGTEXT       12     A
CCGTEXT     *                                                               CCGTEXT       13     A
CCGTEXT     *         NOTE THAT A DIFFERENT VERSION OF *CCGTEXT* EXISTS FOR EACH CCGTEXT   14     A
CCGTEXT     *         HOST COMPILER SINCE IT CONTAINS SYMBOLS DEFINITIONS THAT CCGTEXT     15     A
CCGTEXT     *         ARE HOST DEPENDENT.                                   CCGTEXT       16     A
CCGTEXT     *                                                               CCGTEXT       17     A
CCGTEXT     *         THE TEXT IS DIVIDED UP INTO THE FOLLOWING SECTIONS -  CCGTEXT       18     A
CCGTEXT     *          INSTALLATION DEPENDENT INSTALLATION PARAMETERS.      CCGTEXT       19     A
CCGTEXT     *         STRUCTURED FILED DEFINITION MACROS                    CCGTEXT       20     A
CCGTEXT     *          I/O CODING CONVENTIONS AND MACROS                    CCGTEXT       21     A
CCGTEXT     *          GENERAL MACROS                                       CCGTEXT       22     A
CCGTEXT     *          MACROS FOR DEBUGGING                                 CCGTEXT       23     A
CCGTEXT     *         CCG FIELD DEFINITIONS, EQU"S AND MACROS.              CCGTEXT       24     A
CCGTEXT     *CALL     OPTIONS                                               CCGTEXT       25     A
CCGTEXT     *CALL     COMADEF                                               CCGTEXT       26     A
CCGTEXT     *CALL     FA=DEFS                                               CCGTEXT       27     A
CCGTEXT     *CALL     CCOMGCM                                               CCGTEXT       28     A
CCGTEXT      COUNTS  TITLE     GENERAL MACROS                               CCGTEXT       29     A
CCGTEXT     **        COUNTS - COUNT NUMBER OF NAMES IN A MICRO STRING      CCGTEXT       30     A
CCGTEXT     *                                                               CCGTEXT       31     A
CCGTEXT     * LEN     COUNTS MICNAME                                        CCGTEXT       32     A
CCGTEXT     *                WHERE MICNAME IS A LIST OF NAMES SEPERATED BY COMMAS. CCGTEXT 33     A
CCGTEXT                                                                     CCGTEXT       34     A
CCGTEXT               MACRO  COUNTS,LEN,MICNAM                              CCGTEXT       35     A
CCGTEXT      .1       SET    0                                              CCGTEXT       36     A
CCGTEXT               IFC    NE,/"MICNAM"//,2                               CCGTEXT       37     A

            0         1         2         3         4         5         6         7         8
            12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCGTEXT

```
CCGTEXT            ECHO   1,Z=("MICNAM")                                    CCGTEXT     38    A
CCGTEXT   .1       SET    .1+1                                              CCGTEXT     39    A
CCGTEXT   LEN      EQU    .1                                                CCGTEXT     40    A
CCGTEXT            ENDM                                                     CCGTEXT     41    A
CCGTEXT   MXX+X    SPACE  4,28                                              CCGTEXT     42    A
CCGTEXT  **        MXX+X  - SELECT GREATER (MAXIMUM FUNCTION) OF TWO INTEGERS.  CCGTEXT  43    A
CCGTEXT  *                                                                  CCGTEXT     44    A
CCGTEXT  *         THIS OPDEF COMPARES TWO INTEGERS IN *XJ* AND *XK*, AND PLACES  CCGTEXT  45    A
CCGTEXT  *         THE ALGEBRAICALLY GREATER VALUE IN *XI*.                 CCGTEXT     46    A
CCGTEXT  *                                                                  CCGTEXT     47    A
CCGTEXT  *         MXI    XJ+XK                                             CCGTEXT     48    A
CCGTEXT  *                                                                  CCGTEXT     49    A
CCGTEXT  *         ENTRY  *XI* = RESULT X-REG, CANNOT BE *XJ* OR *XK*       CCGTEXT     50    A
CCGTEXT  *                *XJ*,*XK* = X-REGS CONTAINING INTEGERS TO BE COMPARED  CCGTEXT  51    A
CCGTEXT  *                                                                  CCGTEXT     52    A
CCGTEXT  *         EXIT   (X.LARGER) = 0                                    CCGTEXT     53    A
CCGTEXT  *                (X.SMALLER) = UNCHANGED                           CCGTEXT     54    A
CCGTEXT  *         IF ON ENTRY (XJ)=(XK),                                   CCGTEXT     55    A
CCGTEXT  *            THEN ON EXIT (XJ)=0 AND (XK)=UNCHANGED                CCGTEXT     56    A
CCGTEXT  *                (XI) = MAX(XJ,XK)                                 CCGTEXT     57    A
CCGTEXT  *                                                                  CCGTEXT     58    A
CCGTEXT  *         USES   XI, XJ, XK                                        CCGTEXT     59    A
CCGTEXT                                                                     CCGTEXT     60    A
CCGTEXT            PURGDEF   MXX+X                                          CCGTEXT     61    A
CCGTEXT                                                                     CCGTEXT     62    A
CCGTEXT   MXX+X    OPDEF  I,J,K                                             CCGTEXT     63    A
CCGTEXT    IX.I X.J-X.K                                                     CCGTEXT     64    A
CCGTEXT    AX.I -1                                                          CCGTEXT     65    A
CCGTEXT    BX.J -X.I*X.J                                                    CCGTEXT     66    A
CCGTEXT    BX.K X.I*X.K                                                     CCGTEXT     67    A
CCGTEXT    IX.I X.J+X.K                                                     CCGTEXT     68    A
CCGTEXT   MXX+X    ENDM                                                     CCGTEXT     69    A
CCGTEXT   MXX-X    SPACE  3                                                 CCGTEXT     70    A
CCGTEXT  **        MXX-X - SELECT MINIMUM OF TWO INTEGERS                   CCGTEXT     71    A
CCGTEXT  *                                                                  CCGTEXT     72    A
CCGTEXT  *         MXI    XJ-XK                                             CCGTEXT     73    A
CCGTEXT  *         ENTRY AND EXIT CONDITIONS SAME AS FOR MAX, EXCEPT THAT   CCGTEXT     74    A
CCGTEXT  *         XI = MIN(XJ,XK)                                          CCGTEXT     75    A
CCGTEXT                                                                     CCGTEXT     76    A
CCGTEXT            PURGDEF   MXX-X                                          CCGTEXT     77    A
CCGTEXT                                                                     CCGTEXT     78    A
CCGTEXT   MXX-X    OPDEF  I,J,K                                             CCGTEXT     79    A
CCGTEXT    IX.I X.J-X.K                                                     CCGTEXT     80    A
CCGTEXT    AX.I  -1                                                         CCGTEXT     81    A
CCGTEXT    BX.J X.I*X.J                                                     CCGTEXT     82    A
CCGTEXT    BX.K -X.I*X.K                                                    CCGTEXT     83    A
CCGTEXT    IX.I X.J+X.K                                                     CCGTEXT     84    A
CCGTEXT   MXX-X    ENDM                                                     CCGTEXT     85    A
CCGTEXT   BIT      SPACE  4,8                                               CCGTEXT     86    A
CCGTEXT  **        BIT - SET SYMBOL TO 2**N.                                CCGTEXT     87    A
CCGTEXT  *                                                                  CCGTEXT     88    A
CCGTEXT  *  SYM    BIT       PWR                                            CCGTEXT     89    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCGTEXT

```
CCGTEXT  *                                                          CCGTEXT      90    A
CCGTEXT  *         ENTRY  [SYM] = SYMBOL TO BE SET                  CCGTEXT      91    A
CCGTEXT  *                [PWR] = DESIRED POWER OF 2                CCGTEXT      92    A
CCGTEXT  *                                                          CCGTEXT      93    A
CCGTEXT  *         EXIT   [SYM] = 2**[PWR]                          CCGTEXT      94    A
CCGTEXT                                                             CCGTEXT      95    A
CCGTEXT            PURGMAC   BIT                                    CCGTEXT      96    A
CCGTEXT                                                             CCGTEXT      97    A
CCGTEXT            MACRO  BIT,SYM,PWR                               CCGTEXT      98    A
CCGTEXT   P        DECMIC PWR                                       CCGTEXT      99    A
CCGTEXT   SYM      SET    1S"P"                                     CCGTEXT     100    A
CCGTEXT   BIT      ENDM                                             CCGTEXT     101    A
CCGTEXT   CALL     SPACE  4,8                                       CCGTEXT     102    A
CCGTEXT  **        CALL - CALL A ROUTINE.                           CCGTEXT     103    A
CCGTEXT  *                                                          CCGTEXT     104    A
CCGTEXT  *         GENERATES--    RJ  =X*ROUTINE*.                  CCGTEXT     105    A
CCGTEXT  *                                                          CCGTEXT     106    A
CCGTEXT  *                                                          CCGTEXT     107    A
CCGTEXT  *         CALL      ROUTINE                                CCGTEXT     108    A
CCGTEXT  *                                                          CCGTEXT     109    A
CCGTEXT  *         ENTRY  *ROUTINE* = NAME OF CALLED ROUTINE        CCGTEXT     110    A
CCGTEXT  *                                                          CCGTEXT     111    A
CCGTEXT  *         USES   NONE                                      CCGTEXT     112    A
CCGTEXT                                                             CCGTEXT     113    A
CCGTEXT                                                             CCGTEXT     114    A
CCGTEXT            PURGMAC   CALL                                   CCGTEXT     115    A
CCGTEXT                                                             CCGTEXT     116    A
CCGTEXT   CALL     MACRO  R                                         CCGTEXT     117    A
CCGTEXT    RJ =X_R                                                  CCGTEXT     118    A
CCGTEXT    ENDM                                                     CCGTEXT     119    A
CCGTEXT   ENTRY.   SPACE  4,8                                       CCGTEXT     120    A
CCGTEXT  **        ENTRY. - DEFINE ENTRY POINT AND ITS CONTENTS.    CCGTEXT     121    A
CCGTEXT  *                                                          CCGTEXT     122    A
CCGTEXT  *         REGISTER SNAPS MAY BE OBTAINED AT ENTRY POINTS DEFINED WITH  CCGTEXT  123  A
CCGTEXT  *         THIS MACRO BY USING THE -REGSNAP- DEBUGGING MACRO.  CCGTEXT  124  A
CCGTEXT  *                                                          CCGTEXT     125    A
CCGTEXT  *                                                          CCGTEXT     126    A
CCGTEXT  * NAME    ENTRY. VALUE,C                                   CCGTEXT     127    A
CCGTEXT  *                                                          CCGTEXT     128    A
CCGTEXT  *                *NAME*  = ENTRY POINT NAME                CCGTEXT     129    A
CCGTEXT  *                                                          CCGTEXT     130    A
CCGTEXT  *                *VALUE* = * OR ** OR ANY OTHER LEGAL COMPASS EXPRESSION  CCGTEXT  131  A
CCGTEXT  *                                                          CCGTEXT     132    A
CCGTEXT  *                *C* - ENTRY NAME SUFFIX CHARACTER         CCGTEXT     133    A
CCGTEXT  *                                                          CCGTEXT     134    A
CCGTEXT  *                         *     (FOR LOCAL SUBROUTINES)    EXPANDS TO--  CCGTEXT  135  A
CCGTEXT  *                                NAME  EQ     *+1S17       CCGTEXT     136    A
CCGTEXT  *                                                          CCGTEXT     137    A
CCGTEXT  *                         **    (FOR GLOBAL SUBROUTINES)   EXPANDS TO--  CCGTEXT  138  A
CCGTEXT  *                                      ENTRY  NAME         CCGTEXT     139    A
CCGTEXT  *                                NAME  EQ     *+1S17       CCGTEXT     140    A
CCGTEXT  *                                                          CCGTEXT     141    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCGTEXT

```
CCGTEXT  *                        ELSE  (FOR GLOBAL STORAGE LOCN)  EXPANDS TO--      CCGTEXT     142     A
CCGTEXT  *                                     ENTRY  NAME                           CCGTEXT     143     A
CCGTEXT  *                               NAME  CON    VALUE                          CCGTEXT     144     A
CCGTEXT  *                                                                          CCGTEXT     145     A
CCGTEXT  *         CALLS  NONE                                                      CCGTEXT     146     A
CCGTEXT  *                                                                          CCGTEXT     147     A
CCGTEXT  *         USES   NONE                                                      CCGTEXT     148     A
CCGTEXT                                                                             CCGTEXT     149     A
CCGTEXT                                                                             CCGTEXT     150     A
CCGTEXT            PURGMAC   ENTRY.                                                 CCGTEXT     151     A
CCGTEXT                                                                             CCGTEXT     152     A
CCGTEXT            MACRO  ENTRY.,NAME,VALUE,C                                       CCGTEXT     153     A
CCGTEXT    IFC NE,^VALUE^*^,3                                                       CCGTEXT     154     A
CCGTEXT    ENTRY NAME_C                                                             CCGTEXT     155     A
CCGTEXT   NAME_C BSS 0                                                              CCGTEXT     156     A
CCGTEXT    IFC EQ,^VALUE^**^,2                                                      CCGTEXT     157     A
CCGTEXT  NAME EQ *+1S17                                                             CCGTEXT     158     A
CCGTEXT    SKIP 1                                                                   CCGTEXT     159     A
CCGTEXT  NAME CON VALUE                                                             CCGTEXT     160     A
CCGTEXT  *                          REGISTER SNAP FACILITY.                         CCGTEXT     161     A
CCGTEXT    IF DEF,)NAME,1                                                           CCGTEXT     162     A
CCGTEXT   NAME REG                                                                  CCGTEXT     163     A
CCGTEXT                                                                             CCGTEXT     164     A
CCGTEXT  ENTRY. ENDM                                                                CCGTEXT     165     A
CCGTEXT            SPACE  3                                                         CCGTEXT     166     A
CCGTEXT  **        EQENT - EQUATE SYMBOL, DECLARE ENTRY POINT.                      CCGTEXT     167     A
CCGTEXT  *                                                                          CCGTEXT     168     A
CCGTEXT  *SYM      EQENT  LAB                                                       CCGTEXT     169     A
CCGTEXT  *                                                                          CCGTEXT     170     A
CCGTEXT  *         ENTRY - "SYM" = SYMBOL TO BE MADE AN ENTRY POINT                 CCGTEXT     171     A
CCGTEXT  *                 "LAB" = VALUE OF ENTRY POINT                             CCGTEXT     172     A
CCGTEXT  *                                                                          CCGTEXT     173     A
CCGTEXT  *         GENERATES -                                                      CCGTEXT     174     A
CCGTEXT  *         ENTRY  SYM                                                       CCGTEXT     175     A
CCGTEXT  *SYM      EQU    LAB                                                       CCGTEXT     176     A
CCGTEXT  *                                                                          CCGTEXT     177     A
CCGTEXT  *         CALLS  NONE                                                      CCGTEXT     178     A
CCGTEXT                                                                             CCGTEXT     179     A
CCGTEXT            PURGMAC EQENT                                                    CCGTEXT     180     A
CCGTEXT                                                                             CCGTEXT     181     A
CCGTEXT            MACRO  EQENT,SYM,LAB                                             CCGTEXT     182     A
CCGTEXT            ENTRY  SYM                                                       CCGTEXT     183     A
CCGTEXT   SYM      EQU    LAB                                                       CCGTEXT     184     A
CCGTEXT   EQENT    ENDM                                                             CCGTEXT     185     A
CCGTEXT   MOVE     SPACE  4,8                                                       CCGTEXT     186     A
CCGTEXT  **        MOVE - MOVE DATA BLOCK.                                          CCGTEXT     187     A
CCGTEXT  *                                                                          CCGTEXT     188     A
CCGTEXT  *                                                                          CCGTEXT     189     A
CCGTEXT  *         MOVE      COUNT,FROM,TO                                          CCGTEXT     190     A
CCGTEXT  *                                                                          CCGTEXT     191     A
CCGTEXT  *         ENTRY  *COUNT* = WORD COUNT OF BLOCK TO BE MOVED                 CCGTEXT     192     A
CCGTEXT  *                *FROM*  = ADDRESS OF FIRST WORD OF BLOCK                  CCGTEXT     193     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCGTEXT

```
CCGTEXT  *                *TO*    = ADDRESS OF FIRST WORD OF DESTINATION   CCGTEXT  194  A
CCGTEXT  *                                                                 CCGTEXT  195  A
CCGTEXT  *         USES   X - 1, 2, 3                                      CCGTEXT  196  A
CCGTEXT  *                A - NONE                                         CCGTEXT  197  A
CCGTEXT  *                B - 1                                            CCGTEXT  198  A
CCGTEXT  *                                                                 CCGTEXT  199  A
CCGTEXT  *         CALLS  MVE= (COMCMVE)                                   CCGTEXT  200  A
CCGTEXT                                                                    CCGA018  114  A
CCGTEXT            PURGMAC  MOVE                                           CCGA018  115  A
CCGTEXT                                                                    CCGTEXT  201  A
CCGTEXT                                                                    CCGTEXT  202  A
CCGTEXT   MOVE     MACRO  C,F,T                                            CCGTEXT  203  A
CCGTEXT            R=     X1,C                                             CCGTEXT  204  A
CCGTEXT            R=     X2,F                                             CCGTEXT  205  A
CCGTEXT            R=     X3,T                                             CCGTEXT  206  A
CCGTEXT            IF     -DEF,B1=1,1                                      CCGTEXT  207  A
CCGTEXT            SB1    1                                                CCGTEXT  208  A
CCGTEXT            RJ     =XMVE=                                           CCGTEXT  209  A
CCGTEXT   MOVE     ENDM                                                    CCGTEXT  210  A
CCGTEXT   PLUG     SPACE  4,40                                             CCGTEXT  211  A
CCGTEXT  **        PLUG - MODIFY COMPILER CODE DURING EXECUTION.           CCGTEXT  212  A
CCGTEXT  *                                                                 CCGTEXT  213  A
CCGTEXT  *         SELF-MODIFYING (PLUGGING) CODE MAY SEEMINGLY FAIL ON CYBER  CCGTEXT  214  A
CCGTEXT  *         74 / 6600 AND LARGER MODELS, DUE TO THE HARDWARE INSTRUCTION  CCGTEXT  215  A
CCGTEXT  *         STACK AND 1- OR 2-WORD INSTRUCTION LOOKAHEAD.  BECAUSE OF THIS  CCGTEXT  216  A
CCGTEXT  *         PROBLEM, AND THE UNREADABILITY OF PLUGGED CODE, THE TECHNIQUE  CCGTEXT  217  A
CCGTEXT  *         IS NOT DESIRABLE FOR WIDE USAGE.  EFFICIENCY CONSIDERATIONS  CCGTEXT  218  A
CCGTEXT  *         ARE OCCASIONALLY MORE IMPORTANT, HOWEVER, SO WE FORMALIZE  CCGTEXT  219  A
CCGTEXT  *         PLUGGING WITH THE FOLLOWING MACRO.  IT INCLUDES A SAFEGUARD  CCGTEXT  220  A
CCGTEXT  *         AGAINST THE STACK/LOOKAHEAD PROBLEM, AND IS PREFERABLE TO  CCGTEXT  221  A
CCGTEXT  *         DIRECT CODE FOR READABILITY AND DEBUGGING EASE.         CCGTEXT  222  A
CCGTEXT  *                                                                 CCGTEXT  223  A
CCGTEXT  *         TWO FORMS OF *PLUG* ARE AVAILABLE ...                   CCGTEXT  224  A
CCGTEXT  *                                                                 CCGTEXT  225  A
CCGTEXT  *                                                                 CCGTEXT  226  A
CCGTEXT  *         FORM 1 - PLUGS A CALLER-PROVIDED WORD OF CODE.          CCGTEXT  227  A
CCGTEXT  *                                                                 CCGTEXT  228  A
CCGTEXT  *         PLUG      AT=,FROM=,FREG=,SREG=,VOID=                   CCGTEXT  229  A
CCGTEXT  *                                                                 CCGTEXT  230  A
CCGTEXT  *                                                                 CCGTEXT  231  A
CCGTEXT  *         FORM 2 - FABRICATES AND PLUGS AN -EQ- JUMP.             CCGTEXT  232  A
CCGTEXT  *         RESTRICTION - THE 1-REGISTER METHOD USED FOR FORMING THE -EQ-  CCGTEXT  233  A
CCGTEXT  *         JUMP IS NOT VALID IF *TO* IS .GT. 177777B ABSOLUTE.  THIS IS  CCGTEXT  234  A
CCGTEXT  *         NOT A PROBLEM IN PL/I, SINCE ALL ADDRESSES ARE .LT. 100000B.  CCGTEXT  235  A
CCGTEXT  *                                                                 CCGTEXT  236  A
CCGTEXT  *         PLUG      AT=,TO=,SREG=,VOID=                           CCGTEXT  237  A
CCGTEXT  *                                                                 CCGTEXT  238  A
CCGTEXT  *                                                                 CCGTEXT  239  A
CCGTEXT  *         ENTRY  *AT*   = ADDRESS WHERE PLUG TO BE STORED.        CCGTEXT  240  A
CCGTEXT  *                *FREG* = (FORM 1 ONLY) (OPTIONAL)  NUMBER OF A,X REG  CCGTEXT  241  A
CCGTEXT  *                         TO USE FOR FETCHING CODE WORD.  MUST BE 1-5.  CCGTEXT  242  A
CCGTEXT  *                         USES A1 AND X1 IF OMITTED.  IGNORED IF *FROM*  CCGTEXT  243  A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCGTEXT

```
CCGTEXT  *                        SPECIFIES AN X-REGISTER.                          CCGTEXT      244     A
CCGTEXT  *                 *FROM* = (FORM 1 ONLY)  SOURCE OF CODE WORD TO PLUG.     CCGTEXT      245     A
CCGTEXT  *                         MAY BE AN X-REGISTER CONTAINING THE CODE WORD     CCGTEXT      246     A
CCGTEXT  *                         ( E.G., FROM=X3 ), OR AN ADDRESS EXPRESSION.      CCGTEXT      247     A
CCGTEXT  *                 *SREG* = (OPTIONAL)  NUMBER OF A,X REG TO USE FOR         CCGTEXT      248     A
CCGTEXT  *                         STORING PLUG.  MUST BE 6 OR 7.  USES X6 AND A6    CCGTEXT      249     A
CCGTEXT  *                         IF OMITTED.                                       CCGTEXT      250     A
CCGTEXT  *                 *TO*   = (FORM 2 ONLY)  -EQ- JUMP ADDRESS.                CCGTEXT      251     A
CCGTEXT  *                 *VOID* = (OPTIONAL)                                       CCGTEXT      252     A
CCGTEXT  *                         *NO* = DO NOT GENERATE STACK-VOIDING CODE.        CCGTEXT      253     A
CCGTEXT  *                         ELSE OR OMITTED = GENERATE VOIDING CODE IF        CCGTEXT      254     A
CCGTEXT  *                            "MODEL" IS CYBER 74 / 6600 OR LARGER.          CCGTEXT      255     A
CCGTEXT  *                                                                           CCGTEXT      256     A
CCGTEXT  *         CALLS  NONE                                                       CCGTEXT      257     A
CCGTEXT  *                                                                           CCGTEXT      258     A
CCGTEXT  *         USES   AS ABOVE                                                   CCGTEXT      259     A
CCGTEXT                                                                              CCGTEXT      260     A
CCGTEXT                                                                              CCGTEXT      261     A
CCGTEXT            PURGMAC   PLUG                                                    CCGTEXT      262     A
CCGTEXT                                                                              CCGTEXT      263     A
CCGTEXT   PLUG     MACROE AT,TO,FROM,FREG,SREG,VOID                                  CCGTEXT      264     A
CCGTEXT            LOCAL  FN,SN                                                      CCGTEXT      265     A
CCGTEXT   SN       SET    SREG  6                                                    CCGTEXT      266     A
CCGTEXT            IFLT   SN,6,1                                                     CCGTEXT      267     A
CCGTEXT            ERR    *PLUG* STORE REG MUST BE 6 OR 7 AT ["SEQUENCE"].           CCGTEXT      268     A
CCGTEXT   .F1      IFC    EQ, TO                                                     CCGTEXT      269     A
CCGTEXT   .F1A     IF     REG,FROM                                                   CCGTEXT      270     A
CCGTEXT  *                           FORM 1A - CODE WORD IN AN X-REGISTER.           CCGTEXT      271     A
CCGTEXT   SN       OCTMIC SN,1                                                       CCGTEXT      272     A
CCGTEXT   SN       MICRO  1,, X"SN"                                                  CCGTEXT      273     A
CCGTEXT            IFC    NE, FROM "SN" ,1                                           CCGTEXT      274     A
CCGTEXT            BX.SN  FROM                                                       CCGTEXT      275     A
CCGTEXT   .F1A     ELSE                                                              CCGTEXT      276     A
CCGTEXT  *                           FORM 1B - FETCH CODE WORD FROM *FROM*.          CCGTEXT      277     A
CCGTEXT   FN       SET    FREG  1                                                    CCGTEXT      278     A
CCGTEXT            IFGE   FN,1,2                                                     CCGTEXT      279     A
CCGTEXT            IFLE   FN,5,1                                                     CCGTEXT      280     A
CCGTEXT            SKIP   1                                                          CCGTEXT      281     A
CCGTEXT            ERR    FETCH REG NOT 1-5 "SEQUENCE"                               CCGTEXT      282     A
CCGTEXT            SA.FN  FROM                                                       CCGTEXT      283     A
CCGTEXT            BX.SN  X.FN                                                       CCGTEXT      284     A
CCGTEXT   .F1A     ENDIF                                                             CCGTEXT      285     A
CCGTEXT   .F1      ELSE                                                              CCGTEXT      286     A
CCGTEXT  *                           FORM 2 - FABRICATE -EQ- JUMP TO *TO*.           CCGTEXT      287     A
CCGTEXT            SX.SN  TO                                                         CCGTEXT      288     A
CCGTEXT            LX.SN  2+30D                                                      CCGTEXT      289     A
CCGTEXT  *                           2000BS48+*JPADR*S32                             CCGTEXT      290     A
CCGTEXT            PX.SN  X.SN,B0                                                    CCGTEXT      291     A
CCGTEXT  *                           0400BS48+*JPADR*S30                             CCGTEXT      292     A
CCGTEXT            LX.SN  -2                                                         CCGTEXT      293     A
CCGTEXT   .F1      ENDIF                                                             CCGTEXT      294     A
CCGTEXT  *                           STORE THE PLUG.                                 CCGTEXT      295     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCGTEXT

```
CCGTEXT            SA.SN  AT                                                  CCGTEXT      296     A
CCGTEXT  *                          GENERATE STACK-VOIDING CODE.               CCGTEXT      297     A
CCGTEXT            IFC    NE, VOID NO ,4                                      CCGTEXT      298     A
CCGTEXT            IFGE   CT.CPU,74,3  COMPILE TIME CPU                       CCGTEXT      299     A
CCGTEXT            RJ     *+1                                                 CCGTEXT      300     A
CCGTEXT            EQ     *+1S17                                              CCGTEXT      301     A
CCGTEXT            BSS    0                                                   CCGTEXT      302     A
CCGTEXT   PLUG     ENDM                                                       CCGTEXT      303     A
CCGTEXT   ROUTINE  SPACE  4,8                                                 CCGTEXT      304     A
CCGTEXT  **        ROUTINE - DEFINE LOCAL SUBROUTINE ENTRY POINT.             CCGTEXT      305     A
CCGTEXT  *                                                                    CCGTEXT      306     A
CCGTEXT  *                                                                    CCGTEXT      307     A
CCGTEXT  * NAME    ROUTINE                                                    CCGTEXT      308     A
CCGTEXT  *                                                                    CCGTEXT      309     A
CCGTEXT  *         ENTRY  *NAME* = NAME OF SUBROUTINE.                        CCGTEXT      310     A
CCGTEXT                                                                       CCGTEXT      311     A
CCGTEXT                                                                       CCGTEXT      312     A
CCGTEXT            PURGMAC   ROUTINE                                          CCGTEXT      313     A
CCGTEXT                                                                       CCGTEXT      314     A
CCGTEXT            MACRO  ROUTINE,NAME                                        CCGTEXT      315     A
CCGTEXT   NAME     JP     *+1S17                                              CCGTEXT      316     A
CCGTEXT            NOREF  )NAME                                               CCGTEXT      317     A
CCGTEXT            IF     DEF,)NAME,1                                         CCGTEXT      318     A
CCGTEXT            PRINT  NAME,(* ENTERED*)                                   CCGTEXT      319     A
CCGTEXT   ROUTINE  ENDM                                                       CCGTEXT      320     A
CCGTEXT   SETCORE  SPACE  4,8                                                 CCGTEXT      321     A
CCGTEXT  **        SETCORE - SET BLOCK OF MEMORY TO A GIVEN VALUE.            CCGTEXT      322     A
CCGTEXT  *                                                                    CCGTEXT      323     A
CCGTEXT  *         SETCORE   FWA,LEN                                          CCGTEXT      324     A
CCGTEXT  *                                                                    CCGTEXT      325     A
CCGTEXT  *         ENTRY  *FWA* = BLOCK ADDRESS                               CCGTEXT      326     A
CCGTEXT  *                *LEN* = BLOCK LENGTH                                CCGTEXT      327     A
CCGTEXT  *                (X6)  = VALUE TO SET                                CCGTEXT      328     A
CCGTEXT  *                                                                    CCGTEXT      329     A
CCGTEXT  *         USES   X1, A6                                              CCGTEXT      330     A
CCGTEXT  *                                                                    CCGTEXT      331     A
CCGTEXT  *         CALLS  ISC=  ( CCOMISC )                                   CCGTEXT      332     A
CCGTEXT                                                                       CCGTEXT      333     A
CCGTEXT                                                                       CCGTEXT      334     A
CCGTEXT            PURGMAC   SETCORE                                          CCGTEXT      335     A
CCGTEXT                                                                       CCGTEXT      336     A
CCGTEXT   SETCORE  MACRO  F,L                                                 CCGTEXT      337     A
CCGTEXT            SA6    F                                                   CCGTEXT      338     A
CCGTEXT   .1       IF     DEF,L                                               CCGTEXT      339     A
CCGTEXT   .1       IF     ABS,L                                               CCGTEXT      340     A
CCGTEXT   .1       IF     -REG,L                                              CCGTEXT      341     A
CCGTEXT   .1       IFLT   L,60                                                CCGTEXT      342     A
CCGTEXT   .2       IFNE   L,1                                                 CCGTEXT      343     A
CCGTEXT            MX1    L-1                                                 CCGTEXT      344     A
CCGTEXT            IF     -DEF,B1=1,1                                         CCGTEXT      345     A
CCGTEXT            SB1    1                                                   CCGTEXT      346     A
CCGTEXT  +         LX1    1                                                   CCGTEXT      347     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCGTEXT

```
CCGTEXT            SA6    A6+B1                                          CCGTEXT     348    A
CCGTEXT            MI     X1,*                                           CCGTEXT     349    A
CCGTEXT   .1       ELSE                                                  CCGTEXT     350    A
CCGTEXT            R=     X1,L                                           CCGTEXT     351    A
CCGTEXT            RJ     =XISC=                                         CCGTEXT     352    A
CCGTEXT   .2       ENDIF                                                 CCGTEXT     353    A
CCGTEXT   .1       ENDIF                                                 CCGTEXT     354    A
CCGTEXT   SETCORE  ENDM                                                  CCGTEXT     355    A
CCGTEXT   SETCORE  ENDM                                                  CCGTEXT     356    A
CCGTEXT   SETZERO  SPACE  2                                              CCGTEXT     357    A
CCGTEXT  **        SETZERO - SET A BLOCK OF MEMORY TO ZERO               CCGTEXT     358    A
CCGTEXT  *                                                               CCGTEXT     359    A
CCGTEXT  *         SETZERO  FWA,LEN                                      CCGTEXT     360    A
CCGTEXT  *                                                               CCGTEXT     361    A
CCGTEXT  *         USES   X1, X6, A6 OR CALLS ISC=                       CCGTEXT     362    A
CCGTEXT                                                                  CCGTEXT     363    A
CCGTEXT   SETZERO  MACRO  F,L                                            CCGTEXT     364    A
CCGTEXT            MX6    0                                              CCGTEXT     365    A
CCGTEXT            SETCORE F,L                                           CCGTEXT     366    A
CCGTEXT   SETZERO  ENDM                                                  CCGTEXT     367    A
CCGTEXT  *CALL     CCGDBGM                                               CCGTEXT     368    A
CCGTEXT  *CALL     DBG=MAC                                               CCGTEXT     369    A
CCGTEXT  *CALL     CCGILFD                                               CCGTEXT     370    A
CCGTEXT   BIT      EJECT                                                 CCGTEXT     371    A
CCGTEXT  **        BI. - DEFINE FIRST WORD OF A 2 WORD ENTRY IN *BIT*    CCGTEXT     372    A
CCGTEXT                                                                  CCGTEXT     373    A
CCGTEXT            DESCRIBE BI.,60                                       CCGTEXT     374    A
CCGTEXT   RB       DEFINE 1           REACHABLE BLOCK ( SET FOR PROG BLOCKS ONLY   CCGTEXT     375    A
CCGTEXT   CB       DEFINE 1           CODED ( NOW IN *MCG/SII* FORMAT )             CCGTEXT     376    A
CCGTEXT   IL       DEFINE 1           INITIAL LABEL                                 CCGTEXT     377    A
CCGTEXT   FJ       DEFINE 1           FINAL *UJP*                                   CCGTEXT     378    A
CCGTEXT   LJP      DEFINE 1           BLOCKS CONSISTS OF A *LAB/UJP* ONLY           CCGTEXT     379    A
CCGTEXT   HN       DEFINE 1           =1 IF HEADER NODE OF A INTERVAL & *SCR*       CCGTEXT     380    A
CCGTEXT   FTH      DEFINE 1           =1 IF A *HB* THAT IS FROM A *FTH* LOOP        CCGTEXT     381    A
CCGTEXT   OLN      DEFINE 15          FOR HB ONLY, BN OF HB CONTAINING OUTER LOOP   CCGTEXT     382    A
CCGTEXT   PC       DEFINE 2           PARCEL COUNT                                  CCGTEXT     383    A
CCGTEXT   FLN      DEFINE 18          FIRST LINE NUMBER ( PROG BLOCKS ONLY )        CCGTEXT     384    A
CCGTEXT   BVI      DEFINE 18          *BVT* INDEX TO BIT VECTORS                    CCGTEXT     385    A
CCGTEXT                                                                  CCGTEXT     386    A
CCGTEXT   PII      DEQU   PC,17       INDEX TO *PSI* POST STORE INFO ( "0 )         CCGTEXT     387    A
CCGTEXT   PBN      DEQU   FLN         PROG BLOCK INDEX OF HOLDING BLOCK             CCGTEXT     388    A
CCGTEXT  *                            *BIT* INDEX OF HEADER NODE OF LOOP THAT IT    CCGTEXT     389    A
CCGTEXT  *                            IS THE HOLDING BLOCK OF                       CCGTEXT     390    A
CCGTEXT   MLT      SPACE  3,14                                           CCGTEXT     391    A
CCGTEXT  **        ML. - MOD LIST TABLE ENTRY FORMAT                     CCGTEXT     392    A
CCGTEXT  *                                                               CCGTEXT     393    A
CCGTEXT  *         *MLT* IS THE *MOD* LIST INDEX TABLE.  *MOD* CONSISTS OF RLIST   CCGTEXT     394    A
CCGTEXT  *         INSTRUCTIONS ( 4 WORDS/INSTRUCTION ), AND THE FIRST ENTRY IS    CCGTEXT     395    A
CCGTEXT  *         A DUMMY *BOS*.                                        CCGTEXT     396    A
CCGTEXT  *         OPERANDS OF INSTRUCTIONS IN *MOD* WHICH REFERENCE INSTRUCTIONS  CCGTEXT     397    A
CCGTEXT  *         IN *SEQ* HAVE BIT 15 OF THE R-NUMBER SET ( R+100000B ) .        CCGTEXT     398    A
CCGTEXT  *                                                               CCGTEXT     399    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCGTEXT

```
CCGTEXT  *        FOR FURTHER INFORMATION ONE MAY LOOK AT *MPB* IN *GPO*.          CCGTEXT     400    A
CCGTEXT                                                                             CCGTEXT     401    A
CCGTEXT           DESCRIBE ML.,60    MCW(DEL,II,NI,MTI)                             CCGTEXT     402    A
CCGTEXT   DEL     DEFINE 1           DELETE FLAG                                    CCGTEXT     403    A
CCGTEXT           DEFINE 5                                                          CCGTEXT     404    A
CCGTEXT   II      DEFINE 18          INSERT INDEX ( IN *SEQ* )                      CCGTEXT     405    A
CCGTEXT   NI      DEFINE 18          N. INSTRUCTIONS TO BE INSERTED                 CCGTEXT     406    A
CCGTEXT   MTI     DEFINE 18          *MOD* TABLE INDEX                              CCGTEXT     407    A
CCGTEXT   TET     SPACE  3,14                                                       CCGTEXT     408    A
CCGTEXT  **       T. - TEMP EQUIVALENCE TABLE FORMAT                                CCGTEXT     409    A
CCGTEXT                                                                             CCGTEXT     410    A
CCGTEXT           DESCRIBE T.,60     TETW(FLAGS,ITI,BI,CA)                          CCGTEXT     411    A
CCGTEXT   INV     DEFINE 1           =1 IF *INV* TEMP                               CCGTEXT     412    A
CCGTEXT   EQV     DEFINE 1           EQV TO ANOTHER *TET*                           CCGTEXT     413    A
CCGTEXT   FA      DEFINE 1           FINAL ASSIGNMENT OF *CA* ( *POST* )            CCGTEXT     414    A
CCGTEXT           DEFINE 2                                                          CCGTEXT     415         I
          -CCG21
CCGTEXT           DEFINE 1                                                          CCG21         3         I
          -CCGA049
CCGTEXT   CAL     DEFINE 1           CA FIELD LOCKED                                CCGA049      12    A
CCGTEXT   RF      DEFINE 1           =1 IF SUBSCRIPT USE IN DEFINING LOOP           CCG21         4    A
CCGTEXT   BIP     DEFINE 1           BASE MEMBER OF *IP* CLASS                      CCGTEXT     416    A
CCGTEXT   ITI     DEFINE 18          INDEX TO INC FORMULA INFO IN *IIT* ( ^INV )    CCGTEXT     417    A
CCGTEXT   BI      DEFINE 18          INDEX IN BLOCK WHERE *LD* GOES                 CCGTEXT     418    A
CCGTEXT   CA      DEFINE 18          BIAS OF TST* ( SET IN SQZ * )                  CCGTEXT     419    A
CCGTEXT                                                                             CCGTEXT     420    A
CCGTEXT   HBN     DEQU   ITI         BN OF HB (LOOP) THAT IT. IS ASSOC WITH         CCGTEXT     421    A
CCGTEXT   REG     DEQU   BI          REGISTER THAT IT IS IN ( *MTA* )               CCGTEXT     422    A
CCGTEXT   IPF     DEQU   ITI,19      *IP* FIELDS                                    CCGTEXT     423    A
CCGTEXT   LC.     SPACE  3,12                                                       CCGTEXT     424    A
CCGTEXT  **       LC. - LABEL CHANGE TABLE FOR MULTI PREDECESSOR LOOPS              CCGTEXT     425    A
CCGTEXT                                                                             CCGTEXT     426    A
CCGTEXT           DESCRIBE LC.                                                      CCGTEXT     427    A
CCGTEXT           DEFINE 6                                                          CCGTEXT     428    A
CCGTEXT   NEXT    DEFINE 18          LINK TO NEXT ON CHAIN, 0 IF NONE               CCGTEXT     429    A
CCGTEXT   GLN     DEFINE 18          H FIELD OF GL VALUE OF LABEL                   CCGTEXT     430    A
CCGTEXT   HBN     DEFINE 18          BN OF HB THAT LABEL IS DEFINED IN              CCGTEXT     431    A
CCGTEXT   WC      SPACE  3,14                                                       CCGTEXT     432         I
          -CCG16
CCGTEXT  **       WC. - FORMAT OF WORD C OF SYMTAB FOR ADDRESS DEFINITION           CCGTEXT     433         I
          -CCG16
CCGTEXT                                                                             CCGTEXT     434         I
          -CCG16
CCGTEXT           DESCRIBE WC.                                                      CCGTEXT     435         I
          -CCG16
CCGTEXT   RL      DEFINE 2           0 - ABS, 1 - PROG, 2 - COM, 3 - EXT            CCGTEXT     436         I
          -CCG16
CCGTEXT           DEFINE 13                                                         CCGTEXT     437         I
          -CCG16
CCGTEXT   QBN     DEFINE 12          QUAL BLOCK NUMBER                              CCGTEXT     438         I
          -CCG16
CCGTEXT   RB      DEFINE 9           BLOCK NUMBER ( COMPILER VALUE )                CCGTEXT     439         I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCGTEXT

          -CCG16
CCGTEXT    RA       DEFINE 24          BLOCK/PROGRAM RELATIVE ADDRESS                CCGTEXT      440          I
          -CCG16
CCGTEXT    WRITEP   SPACE  3,14                                                      CCGTEXT      441     A
CCGTEXT   **        WRITEP - WRITE PSEUDO OP WORD TO *SLIST* FILE                   CCGTEXT      442     A
CCGTEXT   *                                                                         CCGTEXT      443     A
CCGTEXT   *         ENTRY  (X6) = LOW 48 BITS OF PSEUDO OP WORD                     CCGTEXT      444     A
CCGTEXT   *                                                                         CCGTEXT      445     A
CCGTEXT   *         WRITEP OPNAME                                                   CCGTEXT      446     A
CCGTEXT                                                                             CCGTEXT      447     A
CCGTEXT    WRITEP   MACRO  OPCD                                                     CCGTEXT      448     A
CCGTEXT             R=     B3,OC.OPCD                                               CCGTEXT      449     A
CCGTEXT             RJ     =XWPW#                                                   CCGTEXT      450     A
CCGTEXT             ENDM                                                            CCGTEXT      451     A
CCGTEXT    HC.      TITLE  HOST DEFINED SYMBOLS AND MICROS                          CCGTEXT      452     A
CCGTEXT   *         GET *HCDEFS* AND *SYMDEFS* FROM THE HOST OLDPL.                 CCGTEXT      453     A
CCGTEXT                                                                             CCGTEXT      454     A
CCGTEXT             QUAL   HOST                                                     CCGTEXT      455     A
CCGTEXT   *CALL     HCDEFS                                                          CCGTEXT      456     A
CCGTEXT             QUAL                                                            CCGTEXT      457     A
CCGTEXT                                                                             CCGTEXT      458     A
CCGTEXT    HC$MCIS  EQU    /HOST/HC$MCIS                                            CCGTEXT      459     A
CCGTEXT                                                                             CCGTEXT      460     A
CCGTEXT   *         HC= - MACRO TO EQUATE HC$ SYMBOLS TO HC. SYMBOLS WHICH CCG      CCGTEXT      461     A
CCGTEXT   *         REFERENCES.                                                     CCGTEXT      462     A
CCGTEXT                                                                             CCGTEXT      463     A
CCGTEXT    HC=      MACRO  A                                                        CCGTEXT      464     A
CCGTEXT             IF     DEF,/HOST/HC$A,2                                         CCGTEXT      465     A
CCGTEXT    HC.A     EQU    /HOST/HC$A                                               CCGTEXT      466     A
CCGTEXT             SKIP   4                                                        CCGTEXT      467     A
CCGTEXT             IF     -MIC,HC$A,2                                              CCGTEXT      468     A
CCGTEXT             ERR    HC$A NOT DEFINED BY THE HOST                             CCGTEXT      469     A
CCGTEXT             SKIP   1                                                        CCGTEXT      470     A
CCGTEXT    HC.A     MICRO  1,,^"HC$A"^                                              CCGTEXT      471     A
CCGTEXT             ENDM                                                            CCGTEXT      472     A
CCGTEXT                                                                             CCGTEXT      473     A
CCGTEXT             HC=    FLB                                                      CCGTEXT      474     A
CCGTEXT             HC=    NXLB                                                     CCGA072        6     A
CCGTEXT             HC=    FPAS                                                     CCGTEXT      475     A
CCGTEXT             HC=    FRTP                                                     CCGTEXT      476     A
CCGTEXT             HC=    HDT                                                      CCGTEXT      477          I
          -CCG16
CCGTEXT             HC=    IA                                                       CCG18          2     A
CCGTEXT             HC=    ID                                                       CCGTEXT      478     A
CCGTEXT             HC=    LNT                                                      CCGTEXT      479     A
CCGTEXT             HC=    MCIS                                                     CCGTEXT      480     A
CCGTEXT             HC=    MCT                                                      CCGTEXT      481          I
          -CCG16
CCGTEXT             HC=    NDAT                                                     CCGTEXT      482          I
          -CCG29
CCGTEXT             HC=    RJXJ                                                     CCGTEXT      483     A
CCGTEXT             HC=    RJTBN                                                    CCGTEXT      484     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCGTEXT

```
CCGTEXT            HC=    RJ6                                                  CCGTEXT      485     A
CCGTEXT            HC=    ROL                                                  CCGTEXT      486     A
CCGTEXT            HC=    STP                                                  CCG16         10     A
CCGTEXT            HC=    UDVB                                                 CCGTEXT      487     A
CCGTEXT                                                                         CCGTEXT      488     A
CCGTEXT            HC=    2OC                                                  CCG16         11     A
CCGTEXT                                                                         CCGTEXT      489     A
CCGTEXT            ECHO   2,Z=(MDL,MODLVL,LPNAME,LPN,VER)                      CCGTEXT      490     A
CCGTEXT            IF     MIC,Z$,1                                             CCGTEXT      491     A
CCGTEXT   Z        MICRO  1,,^"Z"^                                             CCGTEXT      492           I
          -CCG16
CCGTEXT   Z        MICRO  1,,^"Z$"^                                            CCG16         12     A
CCGTEXT  *         GET SYMDEFS                                                 CCGTEXT      493     A
CCGTEXT                                                                         CCGTEXT      494     A
CCGTEXT            SKIP   1                                                    CCGTEXT      495           I
          -CCG16
CCGTEXT  *CALL     SYMDEFS                                                     CCGTEXT      496     A
CCGTEXT   CCG.TM   EJECT                                                       CCGTEXT      497     A
CCGTEXT  **        MISCELLANEOUS CCG MACROS                                    CCGTEXT      498     A
CCGTEXT                                                                         CCGTEXT      499     A
CCGTEXT  **        CCG.SST - MAKE SST CALL FOR *CCG*                           CCGTEXT      500     A
CCGTEXT                                                                         CCGTEXT      501     A
CCGTEXT   CCG.SST  MACRO                                                       CCGTEXT      502     A
CCGTEXT            SST                                                         CCGTEXT      503     A
CCGTEXT            B1=1                                                        CCGTEXT      504     A
CCGTEXT            ENDM                                                        CCGTEXT      505     A
CCGTEXT   SETB1    SPACE  3,14                                                 CCGTEXT      506     A
CCGTEXT  **        SETB1 - SET B1 = 1 IN A HOST CALLABLE ENTRY POINT           CCGTEXT      507     A
CCGTEXT                                                                         CCGTEXT      508     A
CCGTEXT   SETB1    MACRO                                                       CCGTEXT      509     A
CCGTEXT            IFNE   HC.ID,2,1                                            CCGTEXT      510     A
CCGTEXT            SB1    1                                                    CCGTEXT      511     A
CCGTEXT            ENDM                                                        CCGTEXT      512     A
CCGTEXT   ADDWRD   SPACE  3                                                    CCGTEXT      513     A
CCGTEXT  **        ADDWRD - ADD WORD TO MANAGED TABLE.                         CCGTEXT      514     A
CCGTEXT  *                                                                     CCGTEXT      515     A
CCGTEXT  *         ADDWRD TABLE,REG                                            CCGTEXT      516     A
CCGTEXT  *         ENTRY  *TABLE* = TABLE NUMBER.                              CCGTEXT      517     A
CCGTEXT  *                *REG* = REGISTER NAME OR EXPRESSION FOR WORD TO ADD. CCGTEXT      518     A
CCGTEXT  *         USES   A0, X1.                                              CCGTEXT      519     A
CCGTEXT                                                                         CCGTEXT      520     A
CCGTEXT            PURGMAC ADDWRD                                              CCGTEXT      521     A
CCGTEXT   ADDWRD   MACRO  A,B                                                  CCGTEXT      522     A
CCGTEXT            R=     X1,B                                                 CCGTEXT      523     A
CCGTEXT            R=     A0,=XZ$A                                             CCGTEXT      524     A
CCGTEXT            RJ     =XADW$                                               CCGTEXT      525     A
CCGTEXT   ADDWRD   ENDM                                                        CCGTEXT      526     A
CCGTEXT   ALLOC    SPACE  3,14                                                 CCGTEXT      527     A
CCGTEXT  **        ALLOC - ALLOCATE TABLE SPACE.                               CCGTEXT      528     A
CCGTEXT  *                                                                     CCGTEXT      529     A
CCGTEXT  *         ALLOC  TABLE,WORDS                                          CCGTEXT      530     A
CCGTEXT  *         ENTRY  *TABLE* = TABLE NUMBER.                              CCGTEXT      531     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCGTEXT

```
CCGTEXT  *                *WORDS* = WORD COUNT OF TABLE.                    CCGTEXT    532    A
CCGTEXT                                                                      CCGTEXT    533    A
CCGTEXT            PURGMAC ALLOC                                             CCGTEXT    534    A
CCGTEXT   ALLOC    MACRO   A,B                                               CCGTEXT    535    A
CCGTEXT            R=      X1,B                                              CCGTEXT    536    A
CCGTEXT            R=      A0,=XZ$A                                          CCGTEXT    537    A
CCGTEXT            RJ      =XATS$                                            CCGTEXT    538    A
CCGTEXT   ALLOC    ENDM                                                      CCGTEXT    539    A
CCGTEXT   PROCESS  SPACE   3,14                                              CCGTEXT    540    A
CCGTEXT  **        PROCESS - DEFINE PROCESSOR ADDRESSES FOR LIST OF NAMES    CCGTEXT    541    A
CCGTEXT  *                                                                   CCGTEXT    542    A
CCGTEXT  *         PROCESS (A,B,C,...,Z)                                     CCGTEXT    543    A
CCGTEXT                                                                      CCGTEXT    544    A
CCGTEXT            PURGMAC PROCESS                                           CCGTEXT    545    A
CCGTEXT   PROCESS  MACRO   P                                                 CCGTEXT    546    A
CCGTEXT            IRP     P                                                 CCGTEXT    547    A
CCGTEXT            NOREF   .P                                                CCGTEXT    548    A
CCGTEXT   .P       BSS     0                                                 CCGTEXT    549    A
CCGTEXT            IRP                                                       CCGTEXT    550    A
CCGTEXT   PROCESS  ENDM                                                      CCGTEXT    551    A
CCGTEXT   EXT#     SPACE   3,14                                              CCGTEXT    552    A
CCGTEXT  **        EXT# - DECLARE NAMES OF EXTERNALS WITH # SIGNS APPENDED   CCGTEXT    553    A
CCGTEXT  *                                                                   CCGTEXT    554    A
CCGTEXT  *         EXT#    (A,B,...,Z)                                       CCGTEXT    555    A
CCGTEXT                                                                      CCGTEXT    556    A
CCGTEXT   EXT#     MACRO   P                                                 CCGTEXT    557    A
CCGTEXT            IRP     P                                                 CCGTEXT    558    A
CCGTEXT            EXT     P_#                                               CCGTEXT    559    A
CCGTEXT   P        EQU     P_#                                               CCGTEXT    560    A
CCGTEXT            IRP                                                       CCGTEXT    561    A
CCGTEXT            ENDM                                                      CCGTEXT    562    A
CCGTEXT   TABLES   SPACE   3,14                                              CCGTEXT    563    A
CCGTEXT  **        TABLES - DECLARE NAMES OF DYNAMIC TABLES REFERENCED IN A ROUTI  CCGTEXT  564  A
CCGTEXT  *                                                                   CCGTEXT    565    A
CCGTEXT  *         TABLES (TBL1,TBL2,...,TBLN)                               CCGTEXT    566    A
CCGTEXT                                                                      CCGTEXT    567    A
CCGTEXT   TABLES   MACRO   P                                                 CCGTEXT    568    A
CCGTEXT            IRP     P                                                 CCGTEXT    569    A
CCGTEXT            EXT     O$P,L$P                                           CCGTEXT    570    A
CCGTEXT   O.P      EQU     O$P                                               CCGTEXT    571    A
CCGTEXT   L.P      EQU     L$P                                               CCGTEXT    572    A
CCGTEXT            IRP                                                       CCGTEXT    573    A
CCGTEXT            EXT     F$RDT,S$IT,O$SYM                                  CCGTEXT    574    A
CCGTEXT   O.SYM    EQU     O$SYM                                             CCGTEXT    575    A
CCGTEXT   F.RDT    EQU     F$RDT                                             CCGTEXT    576    A
CCGTEXT   IT.      EQU     S$IT                                              CCGTEXT    577    A
CCGTEXT   TABLES   ENDM                                                      CCGTEXT    578    A
CCGTEXT                                                                      CCGTEXT    579    A
CCGTEXT   .Z       SET     1S15                                              CCGTEXT    580    A
CCGTEXT            ECHO    2,P=("HC.FRTP",SS)                                CCGTEXT    581    A
CCGTEXT   I.P      EQU     .Z                                                CCGTEXT    582    A
CCGTEXT   .Z       SET     .Z+1S15                                           CCGTEXT    583    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
         LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCGTEXT

CCGTEXT                                                                  CCGTEXT      584     A
CCGTEXT   HC.NFLB  COUNTS HC.FLB                                         CCGTEXT      585     A
CCGTEXT   WB.CXP   = WB.EXTP                                             CCGTEXT      586     A
CCGTEXT   WB.CXL   = 2                                                   CCGTEXT      587     A
CCGTEXT   EQU-S    SPACE  3,14                                           CCGTEXT      588     A
CCGTEXT  *         DEFINITIONS OF SYMBOLS THAT DO NOT CHANGE             CCGTEXT      589     A
CCGTEXT                                                                  CCGTEXT      590     A
CCGTEXT   WA.LCOP  EQU    0          LABEL CHANGE ORDINAL FIELD          CCGTEXT      591     A
CCGTEXT   WA.LCOL  EQU    12                                             CCGTEXT      592     A
CCGTEXT   WA.BINP  EQU    0          BITS IN NAME                        CCGTEXT      593     A
CCGTEXT   WA.BINL  EQU    12                                             CCGTEXT      594     A
CCGTEXT                                                                  CCGTEXT      595     A
CCGTEXT   HC.TS    EQU    0                                              CCGTEXT      596     A
CCGTEXT   HC.QBT   EQU    0                                              CCGTEXT      597     A
CCGTEXT                                                                  CCGTEXT      598     A
CCGTEXT            END                                                   CCGTEXT      599     A
CCGTEXT  *WEOR                                                           CCGTEXT      600         I
         -CCGA054

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - CCGTEXT

         IDENTIFIER                TOTAL    ACTIVE

         CCGTEXT                     600       584
         CCG16                         3         3
         CCG18                         1         1
         CCG21                         2         1
         CCGA018                       2         2
         CCGA049                       1         1
         CCGA072                       1         1

         LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN WEOR2*

WEOR2*   *DECK WEOR2*                                                    CCGA054        7     A
WEOR2*   *WEOR 0                                                         CCGA054        8     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - WEOR2*

         IDENTIFIER                TOTAL    ACTIVE

         CCGA054                       2         2

         LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN *CCG*

*CCG*    *DECK    *CCG*                                                  *CCG*          1     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - *CCG*

         IDENTIFIER                TOTAL    ACTIVE

         *CCG*                         1         1

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN *CCG*

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA     *DECK     CGIA                                                    CGIA          1     A
CGIA               IDENT  CGIA                                             CGIA          2     A
CGIA      CGIA     TITLE  CGIA - CODE GENERATOR INTERNAL ASSEMBLER         CGIA          3     A
CGIA      B=CGIA   RPVDEF                                                  CGIA          4     A
CGIA     *CALL     SSTCALL                                                 CGIA          5     A
CGIA      CGIA     SPACE  3                                                CGIA          6     A
CGIA     **        CGIA - CODE GENERATOR INTERNAL ASSEMBLER.               CGIA          7     A
CGIA     *                S.I. JASIK - CDC - OCTOBER 1976                  CGIA          8     A
CGIA     *                                                                 CGIA          9     A
CGIA     *         FUNCTIONS -                                             CGIA         10     A
CGIA     *                CONVERT THE CG IL INTO RELOCATABLE BINARY        CGIA         11     A
CGIA     *                PRODUCE A OBJECT LISTING ( O-LIST SELECTED )     CGIA         12     A
CGIA     *         OR     PRODUCE A FILE OF BCD LINE IMAGES THAT CAN BE ASSEMBLED  CGIA  13     A
CGIA     *                BY *COMPASS*.                                    CGIA         14     A
CGIA     *                                                                 CGIA         15     A
CGIA     **        INPUT TO *CGIA* CONSISTS OF THE FOLLOWING -             CGIA         16     A
CGIA     *                                                                 CGIA         17     A
CGIA     *         TEXT STREAM - A WORD ORIENTED STREAM OF OPCODES AND OPERANDS  CGIA  18     A
CGIA     *         WHICH IS SUBDIVIDED INTO 3 SECTIONS, THE INITIAL DECLARATIVES,  CGIA  19     A
CGIA     *         THE PROGRAM BODY WHICH IS MOSTLY MACHINE INSTRUCTIONS, AND  CGIA  20     A
CGIA     *         THE FINAL DECLARATIVES.                                 CGIA         21     A
CGIA     *                                                                 CGIA         22     A
CGIA     *         SYMBOL TABLE - CONTAINS THE NAMES OF THE REFERENCED SYMBOLS  CGIA  23     A
CGIA     *         AND ADDRESS RELOCATION INFORMATION. BECAUSE *CGIA* IS A 1 PASS  CGIA  24     A
CGIA     *         ASSEMBLER ALL ADDRESS"S WILL HAVE BEEN DEFINED PRIOR TO  CGIA  25     A
CGIA     *         CALLING IT. THE SYMTAB FORMAT IS A FUNCTION OF THE HOST  CGIA  26     A
CGIA     *         COMPILER.                                               CGIA         27     A
CGIA     *                                                                 CGIA         28     A
CGIA     *         BLOCK TABLES -                                          CGIA         29     A
CGIA     *          COMMON BLOCK TABLE ( 60/0L_BLOCK-NAME,2/LVL,34/0,24/LENGTH )  CGIA  30     A
CGIA     *                            LVL = 2 OR 3 IMPLIES THE BLOCK IS ECS/LCM.  CGIA  31     A
CGIA     *          LOCAL BLOCK TABLE ( 24/0,18/PC,18/LEN )               CGIA         32     A
CGIA     *                DO NOT INCLUDE ENTRIES FOR *SUB* BLOCKS HERE.   CGIA         33     A
CGIA     *         QUAL BLOCK NAME TABLE ( 42/0L_QUAL-NAME,18/6*N.CHARS IN NAME)  CGIA  34     A
CGIA     *                THE FIRST 2 ENTRIES IN *QBT* ARE 0 AND * .       CGIA         35     A
CGIA     *                                                                 CGIA         36     A
CGIA     *         THE DECK *MACROS* CONTAINS THE LANGUAGE PROCESSOR ASSEMBLY  CGIA  37     A
CGIA     *         TIME MACRO DEFINITIONS IN A FORMAT DEFINED IN IT.       CGIA         38     A
CGIA      ILFMT    SPACE  3,14                                             CGIA         39     A
CGIA     **        ASSEMBLER INPUT FILE FORMATS                            CGIA         40     A
CGIA     *                                                                 CGIA         41     A
CGIA     *         1) MACHINE INSTRUCTIONS                                 CGIA         42     A
CGIA     *                SI. FORMAT FOR NON-TS MODE AND *TS* LONGFILE FORMAT  CGIA  43     A
CGIA     *                WHEN *CGIA* IS BEING ASSEMBLED FOR *TS*.        CGIA         44     A
CGIA     *         2) PSEUDO INSTRUCTIONS                                  CGIA         45     A
CGIA     *                FORMAT IS THAT OF A TYPE IV *IL* INSTRUCTION -  CGIA         46     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA     *                12/P(OC),18/CA,12/H2,18/IH       (  OC  CA,IH,H2 )          CGIA          47     A
CGIA     *                OPTIONAL DATA WORDS MAY FOLLOW .                            CGIA          48     A
CGIA     *                                                                            CGIA          49     A
CGIA     *          3) MACROS                                                         CGIA          50     A
CGIA     *                THE MACRO CALL  * OC P1,P2,... * IS FORMATTED AS            CGIA          51     A
CGIA     *                12/XOR(P(OC),7777B),24/P2,24/P1   ( OC \ 128 )              CGIA          52     A
CGIA     *                12/0,24/P4,24/P3                                            CGIA          53     A
CGIA     *                12/0,24/P6,24/P5                                            CGIA          54     A
CGIA     *                                                                            CGIA          55     A
CGIA     *                THE NUMBER OF PARAMETERS IN A MACRO CALL MUST AGREE         CGIA          56     A
CGIA     *                WITH THE NUMBER IN THE DEFINITION.                          CGIA          57     A
CGIA      PSEUDOS   SPACE  3,20                                                        CGIA          58     A
CGIA     *          PSEUDO INSTRUCTIONS AND HARDWIRED MACROS PROCESSED BY *CGIA*      CGIA          59     A
CGIA     *                                                                            CGIA          60     A
CGIA     *          SY - SYMTAB ORDINAL                                               CGIA          61     A
CGIA     *          WC - WORD COUNT                                                   CGIA          62     A
CGIA     *          A * PRIOR TO WC OR CA MEANS THAT THE FIELD IS 24 BITS.            CGIA          63     A
CGIA     *                                                                            CGIA          64     A
CGIA     *          OPCODE CA,IH       OTHER PARAMETERS / COMMENTS                    CGIA          65     A
CGIA     *                                                                            CGIA          66     A
CGIA     *                             INITIAL GROUP ( MUST OCCUR FIRST )             CGIA          67     A
CGIA     *          LCC       WC       BCD STRING OF WC WORDS IN *C* FORMAT           CGIA          68     A
CGIA     *          IDENT     SY                                                      CGIA          69     A
CGIA     *          TITLE     WC       BCD STRING IN *H* OR *C* FORMAT                CGIA          70     A
CGIA     *          COMNT     WC       BCD STRING IN *H* FORMAT / GOES IN 77 TBL      CGIA          71     A
CGIA     *          LIB    ___         12/P(OC.LIB),48/0L_NAME                        CGIA          72     A
CGIA     *          USBLK              / USEBLK TERMINATES THE INITIAL GROUP.         CGIA          73     A
CGIA     *                             THE *USBLK* MACRO DEF IN *XXXMAC* CONTAINS     CGIA          74     A
CGIA     *                             A *SST* DIRECTIVE AND IT DEFINES THE           CGIA          75     A
CGIA     *                             ORDER OF THE FIXED LOCAL BLOCKS.               CGIA          76     A
CGIA     *                                                                            CGIA          77     A
CGIA     *          USE       BN       / BN = 0 - N FOR COMMON BLOCKS                 CGIA          78          I
         -CCGA005
CGIA     *          USE    BN,CLF      /  B = 0 - N FOR COMMON BLOCKS                 CCGA005        5     A
CGIA     *                                  1S15+LOCAL-BLOCK-ORD FOR LOCAL BLOCKS      CGIA          79     A
CGIA     *                             CLF = 1 INDICATES CONDITIONAL LOADING           CCGA005        6     A
CGIA     *          QUAL      QBO      / QBO = 0     FOR QUAL 0                       CGIA          80     A
CGIA     *                                     1 FOR QUAL *                           CGIA          81     A
CGIA     *                                     2 - N FOR QUAL BLKNAM                  CGIA          82     A
CGIA     *          BSS    SY,*WC      /  SY BSS  *WC                                 CGIA          83     A
CGIA     *          BOS    LN          LN = LINE NUMBER                               CGIA          84     A
CGIA     *                             H2 FIELD MAY CONTAIN LIST FLAGS                CGIA          85     A
CGIA     *                             H2 = 0 NO CHANGE                               CGIA          86     A
CGIA     *                                  1        LIST ON                          CGIA          87     A
CGIA     *                                  2        LIST OFF                         CGIA          88     A
CGIA     *                             / CAUSES PRINTING OF A COMMENT'* LINE LN'       CGIA          89     A
CGIA     *          CON    SY,*CA      / FORMS CON SY+*CA                             CGIA          90     A
CGIA     *                                                                            CGIA          91     A
CGIA     *          DATA   WC,CC       WC WORDS OF BINARY DATA FOLLOW                 CGIA          92     A
CGIA     *                             IF WC = 0 THEN *CC* IS THE DATA ITEM            CGIA          93     A
CGIA     *          DIS       WC       WC WORDS OF BCD DATA FOLLOW IN *H* FORMAT.     CGIA          94     A
CGIA     *          END    CA,SY       / CA = ORD(LABEL FIELD), SY = ORD(XFER)        CGIA          95     A

         0        1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA     *         HOL    CC,F        1 WORD OF BCD DATA IN H FORMAT FOLLOWS          CGIA        96     A
CGIA     *                            CC = CHAR COUNT ( @ 10 ) , F = 1RH,1RL,1RR      CGIA        97     A
CGIA     *         ORG    SY,*CA      / SEE *STD-MAC* FOR DEF OF *ORG* AND *REPI*     CGIA        98     A
CGIA     *         REPI   DLEN,RC,INC,DESTIN                                          CGIA        99     A
CGIA     *                            SECOND WORD IS 24/,18/DESTIN,18/INC             CGIA       100     A
CGIA     *                                                                            CGIA       101     A
CGIA     *         BMI                 / BEGIN *TS* FORMAT MACHINE INSTRUCTION        CGIA       102     A
CGIA     *                            PROCESSING                                      CGIA       103     A
CGIA     *         DCS    CC,I        AND 1 WORD FOLLOWS IN *L* FORMAT                CGIA       104     A
CGIA     *                            / DEFINE CHAR STRING FOR USE AS A MACRO ARG     CGIA       105     A
CGIA     *                            MI   MICRO  1,CC, WORD                          CGIA       106     A
CGIA     *         VFDP   BN,LEN      1 WORD OF BCD DATA IN *L* FORMAT FOLLOWS.       CGIA       107     A
CGIA     *                            BN = POSITION COUNTER - 1                       CGIA       108     A
CGIA     *                                                                            CGIA       109     A
CGIA     *         NOT DESCRIBED - APL,IOM                                            CGIA       110     A
CGIA     *         SUB    CA,FPN      / FPN = FORMAL PARAM NUMBER, CA = XIJ           CGIA       111     A
CGIA     *                            X = 0/1 FOR READ WRITE , I,J = ADDR,DATA        CGIA       112     A
CGIA     *                            REGISTER NUMBERS                                CGIA       113     A
CGIA     *                                                                            CGIA       114     A
CGIA     *         ENTRY AND EXT PSEUDOS ARE NOT NECESSARY SINCE THEY ARE PUT         CGIA       115     A
CGIA     *         OUT BY THE ASSEMBLER WHEN IT INITIALIZES THE SYMBOL TABLE.         CGIA       116     A
CGIA      TABLES   TITLE  ASSEMBLER TABLE FORMATS                                     CGIA       117     A
CGIA     **        FORMAT OF WORD 3 OF *SYMTAB* DURING ASSEMBLY -                     CGIA       118     A
CGIA     *                                                                            CGIA       119     A
CGIA     *         AR. - ADDRESS RELOCATION WORD FORMAT                               CGIA       120     A
CGIA                                                                                  CGIA       121     A
CGIA              DESCRIBE AR.       FOR ^EXTERNALS                                    CGIA       122     A
CGIA      RL      DEFINE 2           0 - ABS, 1 - PROG, 2 - COM, 3 - EXT               CGIA       123     A
CGIA              DEFINE 13                                                           CGIA       124     A
CGIA      QBN     DEFINE 12          QUAL BLOCK NUMBER                                 CGIA       125     A
CGIA      RB      DEFINE 9           BLOCK NUMBER ( COMPILER VALUE )                   CGIA       126     A
CGIA      RA      DEFINE 24          PROGRAM RELATIVE ADDRESS                          CGIA       127     A
CGIA                                                                                  CGIA       128     A
CGIA              DESCRIBE AR.       FOR EXTERNALS                                     CGIA       129     A
CGIA      RL      DEFINE 2           =3                                                CGIA       130     A
CGIA              DEFINE 22                                                           CGIA       131     A
CGIA      XCI     DEFINE 18          EXTERNAL CHAIN INDEX                              CGIA       132     A
CGIA      XLNK    DEFINE 18          XLINK CHAIN INDEX TO *XLT*                        CGIA       133     A
CGIA      AD.     SPACE  2,10                                                          CGIA       134     A
CGIA     **       AD. - LBIT, CBT ADDRESS DEF WORD FORMAT                            CGIA       135     A
CGIA                                                                                  CGIA       136     A
CGIA              DESCRIBE AD.       AD(NPR,RB,ORGC)                                   CGIA       137     A
CGIA              DEFINE 2                                                            CGIA       138     A
CGIA      NPR     DEFINE 10          NUMBER OF PARCELS REMAINING IN CURRENT WORD       CGIA       139     A
CGIA      LCM     DEFINE 1           =1 IF BLOCK IS IN LCM/ECS                         CGIA       140     A
CGIA              DEFINE 14                                                           CGIA       141          I
         -CCGA005
CGIA              DEFINE 13                                                           CCGA005      7     A
CGIA      CL      DEFINE 1                                                            CCGA005      8     A
CGIA      RB      DEFINE 9           BLOCK NUMBER ( LOADER )                           CGIA       142     A
CGIA      ORGC    DEFINE 24          ORG COUNTER                                       CGIA       143     A
CGIA                                                                                  CGIA       144     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA      RBOC     DEQU   ORGC,33     RB AND ORGC FIELDS                          CGIA        145     A
CGIA      CRBO     DEQU   ORGC,34     CL, RV AND ORGC FIELDS                      CCGA005       9     A
CGIA      BLKTBLS  SPACE  2,10                                                    CGIA        146     A
CGIA     **        PIDL - BLOCK NAME AND LENGTH TABLE                            CGIA        147     A
CGIA     *                                                                       CGIA        148     A
CGIA     *         42/0L_NAME,1/ECS,17/LEN OR LEN/8 IF ECS=1                     CGIA        149     A
CGIA      LBT      SPACE  2,5                                                    CGIA        150     A
CGIA     **        LBT - LOCAL BLOCK ADDRESS TABLE                               CGIA        151     A
CGIA     *                                                                       CGIA        152     A
CGIA     *         42/,18/LEN WHICH IS CHANGED TO 42/,18/FWA BY *CG.RBT*         CGIA        153     A
CGIA      LBIT     SPACE  2,6                                                    CGIA        154     A
CGIA     **        LBIT - LOCAL BLOCK INFO TABLE , 2 WORDS PER ENTRY FOR EACH    CGIA        155     A
CGIA     *         LOCAL BLOCK ( NOT INCLUDING *SUB* BLOCKS ) .                  CGIA        156     A
CGIA     *                                                                       CGIA        157     A
CGIA     *         AD WORD 12/P(PC),15/,9/RB,24/ORGC  ( RB = 1 ALWAYS )          CGIA        158     A
CGIA     *                45/DATA IN LAST PARTIAL WORD,11/0,4/LAST RELOC BYTE    CGIA        159     A
CGIA      CBT      SPACE  2,5                                                    CGIA        160     A
CGIA     **        CBT - COMMON BLOCK INFO TABLE , 2 WORDS PER ENTRY             CGIA        161     A
CGIA     *                                                                       CGIA        162     A
CGIA     *         AD WORD 12/P(PC),15/0,9/RB,24/ORGC                            CGIA        163     A
CGIA     *                42/0,18/LOADER BLOCK NUMBER                            CGIA        164     A
CGIA      LAFT     SPACE  2,12                                                   CGIA        165     A
CGIA     **        LAFT - LINK AND FILL CHAIN TABLE                              CGIA        166     A
CGIA     *                                                                       CGIA        167     A
CGIA     *         12/,18/LBN OR EXT NUM,30/LINK OR FILL BYTE                    CGIA        168     A
CGIA     *         LINK/FILL BYTE FORMAT IS 1/1,2/RELOC,9/RB,18/ORGC             CGIA        169     A
CGIA     *         RELOC = 0/1/2 FOR L/M/U RELOCATION AND RB IS USUALLY = 1      CGIA        170     A
CGIA      XFT      SPACE  2,5                                                    CGIA        171     A
CGIA     **        XFT - XFILL BYTE TABLE                                        CGIA        172     A
CGIA     *                                                                       CGIA        173     A
CGIA     *         6/0,24/ORGC,6/LOWBIT,6/LEN,9/RB OF ADDR,9/RB OF DESTIN ( =1 )  CGIA        174     A
CGIA      XLT      SPACE  2,14                                                   CGIA        175     A
CGIA     **        XLT - XLINK BYTE TABLE ( MAX LENGTH = 777B )                  CGIA        176     A
CGIA     *                                                                       CGIA        177     A
CGIA     *         6/0,24/ORGC,6/LOWBIT,6/LEN,9/LINK TO PREV,9/RB OR DESTIN ( =1 CGIA        178     A
CGIA      FPIT     SPACE  2,6                                                    CGIA        179     A
CGIA     **        FPT - FORMAL PARAMETER INFO TABLE,  1 WD/F.P.                 CGIA        180     A
CGIA     *                                                                       CGIA        181     A
CGIA     *         24/0,18/SYMTAB INDEX,18/INDEX TO SUBTABLE IN *FPST*           CGIA        182     A
CGIA     *         FPST HOLDS THE SUB WORDS OF THE F.P.S IN THE CORRECT ORDER.   CGIA        183     A
CGIA      LNT      SPACE  2,6                                                    CGIA        184     A
CGIA     **        LNT - LINE NUMBER TABLE FOR *FID*                             CGIA        185     A
CGIA     *                                                                       CGIA        186     A
CGIA     *         HEADER WORD - 12/5700B,12/WC,12/HC.ID,24/0                    CGIA        187     A
CGIA     *         TABLE WORDS - 18/LABEL IN BINARY,6/0,18/LINE NUMBER,18/FWA    CGIA        188     A
CGIA               SPACE  2                                                      CGIA        189     A
CGIA               TABLES (SYM,PIDL,LBT,LBIT,LCC,ENTR,CBT,LAFT,XFT,LIB)          CGIA        190         I
         -CCG16
CGIA                                                                             CGIA        191         I
         -CCG16
CGIA               ECHO   2,Z=(QBT,LNT)                                          CGIA        192         I
         -CCG16

         0        1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA              IFNE   HC.Z,0,1                                                CGIA        193          I
          -CCG16
CGIA              TABLES Z                                                       CGIA        194          I
          -CCG16
CGIA                                                                             CGIA        195          I
          -CCG16
CGIA              IFNE   HC.FPAS,0,1                                              CGIA        196          I
          -CCG16
CGIA              TABLES (FPT,FPST)                                              CGIA        197          I
          -CCG16
CGIA                                                                             CGIA        198          I
          -CCG16
CGIA              EXT    F.SLST,F.LGO,F.CMPS                                     CGIA        199          I
          -CCG16
CGIA              ECHO   2,A=(PBUF,O.GLT,O.SST),B=(PBUF#,O$GLT,O$SST)            CGIA        200          I
          -CCG16
CGIA              EXT    B                                                       CGIA        201          I
          -CCG16
CGIA       A      EQU    B                                                       CGIA        202          I
          -CCG16
CGIA      **      TABLE - DEFINE ORDER OF ASSEMBLER TABLES                       CCG16        13     A
CGIA                                                                             CCG16        14     A
CGIA              MACRO  TABLE,N                                                 CCG16        15     A
CGIA       Z$N    EQU    .T                                                      CCG16        16     A
CGIA       O.N    EQU    O.BLK+.T                                                CCG16        17     A
CGIA       L.N    EQU    L.BLK+.T                                                CCG16        18     A
CGIA       .T     SET    .T+1                                                    CCG16        19     A
CGIA              ENDM                                                           CCG16        20     A
CGIA       .T     SET    1                                                       CCG16        21     A
CGIA                                                                             CCG16        22     A
CGIA              TABLES (BLK,SYM,CBT)  AND LBT IN CRADLE                        CCG16        23          I
          -CCG29
CGIA              TABLES (BLK,SYM,CBT,CUT)   AND *LBT* IN CRADLE                 CCG29         3     A
CGIA                                                                             CCG16        24     A
CGIA              IFNE   HC.FPAS,0,1                                             CCG16        25     A
CGIA       FPST   TABLE                                                          CCG16        26     A
CGIA              ECHO   1,Z=(LAFT,XFT,XLT,LNT,PIDL,LBIT,ENTR,LCC,LIB)           CCG16        27     A
CGIA       Z      TABLE                                                          CCG16        28     A
CGIA                                                                             CCG16        29     A
CGIA              EXT    F.SLST,F.LGO,F.CMPS,PBUF#                               CCG16        30     A
CGIA       PBUF   EQU    PBUF#                                                   CCG16        31     A
CGIA       MXB    SPACE  3,10                                                    CGIA        203     A
CGIA      **      MXB - VARIABLE MASK OPDEF                                      CGIA        204     A
CGIA      *                                                                      CGIA        205     A
CGIA      *       MXI    BJ          PRODUCES A MASK (BJ)+1 BITS LONG IN XI      CGIA        206     A
CGIA                                                                             CGIA        207     A
CGIA       MXB    OPDEF  I,J                                                     CGIA        208     A
CGIA              MX.I   1                                                       CGIA        209     A
CGIA              AX.I   B.J                                                     CGIA        210     A
CGIA              ENDM                                                           CGIA        211     A
CGIA       DECLS  TITLE  LOCAL STORAGE AND DECLARATIVES                          CGIA        212     A
CGIA      **      OBSS - MACRO TO DECLARE BLANK STORAGE SPACE SO THAT IT         CGIA        213          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
          -CCG16
CGIA      *         OVERLAPS THE SPACE OCCUPIED BY THE END PROCESSOR IN NON   CGIA      214      I
          -CCG16
CGIA      *         TS MODE.                                                  CGIA      215      I
          -CCG16
CGIA      *                                                                   CGIA      216      I
          -CCG16
CGIA      * SYM     OBSS   AEXP                                               CGIA      217      I
          -CCG16
CGIA                                                                          CGIA      218      I
          -CCG16
CGIA                MACRO  OBSS,SY,N                                          CGIA      219      I
          -CCG16
CGIA                IFNE   HC.TS,0,2                                          CGIA      220      I
          -CCG16
CGIA       SY       BSS    N                                                  CGIA      221      I
          -CCG16
CGIA                SKIP   2                                                  CGIA      222      I
          -CCG16
CGIA       SY       =      L.WII                                              CGIA      223      I
          -CCG16
CGIA       L.WII    SET    L.WII+N                                            CGIA      224      I
          -CCG16
CGIA                ENDM                                                      CGIA      225      I
          -CCG16
CGIA                                                                          CGIA      226      I
          -CCG16
CGIA                USE    WII                                                CGIA      227      I
          -CCG16
CGIA                NOREF  F.WII,L.WII                                        CGIA      228      I
          -CCG16
CGIA       F.WII    BSS    0           FWA OF END PROCESSOR BLOCK             CGIA      229      I
          -CCG16
CGIA                USE    *                                                  CGIA      230      I
          -CCG16
CGIA       L.WII    SET    F.WII       NEXT AVAILABLE ADDRESS IN WII BLOCK    CGIA      231      I
          -CCG16
CGIA                SPACE  2                                                  CGIA      232      I
          -CCG16
CGIA       OBSS     OPSYN  BSS         DEFUNCT MACRO                          CCG16      32     A
CGIA                                                                          CCG16      33     A
CGIA      *         LOCAL STORAGE                                             CGIA      233     A
CGIA                                                                          CGIA      234     A
CGIA       BCDC     OBSS   1           "0 IF OL ! C  , < 0 IF C               CGIA      235     A
CGIA       CCB      DATA   2           COMPILER COMMON BLOCK BIAS             CGIA      236     A
CGIA      *                           =3 IF //COM IN PIDL, ELSE =2            CGIA      237     A
CGIA       IL.LAF   OBSS   1           INIT LEN OF *LAF* ( (N.EXT+1)/2+1 )    CGIA      238     A
CGIA       NAE      BSSZ   1           N. ASSEMBLY ERRORS                     CGIA      239     A
CGIA                DATA   28L ASSEMBLY ERRORS IN PROGNAM                     CGIA      240     A
CGIA       N.ENT    BSS    1           N. ENTRY POINTS ( SET BY CG.ASN )      CGIA      241     A
CGIA       XCI      CON    0           EXTERNAL CHAIN INDEX                   CGIA      242     A
CGIA       UBO      DATA   1S15,1S15   CURRENT LAST USE BLOCK ORDINALS         CGIA      243     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                                                                                  CGIA        244     A
CGIA      NLF      DATA   0           NO LIST FLAG FOR FORCEING UPPER, ETC           CGIA        245     A
CGIA      LBUF.WC  OBSS   1           N.WORDS IN LBUF                                CGIA        246     A
CGIA      OBUF     OBSS   12  4+8     ADDRESS, OCTAL CODE, RELOC INFO  AND           CGIA        247     A
CGIA      LBUF     =      OBUF+4      BCD LINE IMAGE BUFFER                          CGIA        248     A
CGIA                                                                                  CGIA        249     A
CGIA      MIW      BSSZ   2           MACRO INFO WORDS ( INDEX AND HEADER )          CGIA        250     A
CGIA                                                                                  CGIA        251     A
CGIA      L.MP     EQU    24          LENGTH OF A MACRO PARAMETER FIELD              CGIA        252     A
CGIA      L.MRB    EQU    12          LENGTH OF A MACRO RELOC BYTE                   CGIA        253     A
CGIA               IFEQ   HC.TS,0                                                    CGIA        254         I
          -CCG16
CGIA      L.IBUF   EQU    100B        INPUT BUFFER LENGTH                            CGIA        255         I
          -CCG16
CGIA               SKIP   1                                                          CGIA        256         I
          -CCG16
CGIA      L.IBUF   EQU    40B                                                       CGIA        257         I
          -CCG16
CGIA      L.IBUF   EQU    40B         LENGTH OF INPUT BUFFER                         CCG16        34     A
CGIA                                                                                  CGIA        258     A
CGIA      IBUF     OBSS   L.IBUF                                                     CGIA        259     A
CGIA      IBUFL    OBSS   1           12/P(EOB),48/0  END OF BUFFER SENTINAL         CGIA        260     A
CGIA      EOBA     =      IBUF+L.IBUF-1  END OF BUFFER ADDRESS                       CGIA        261     A
CGIA                                                                                  CCG16        35     A
CGIA               ENTRY  F$SST                                                      CCG16        36     A
CGIA      F$SST    BSSZ   8           SET SYMBOL TABLE                               CCG16        37     A
CGIA               SPACE  3,14                                                       CGIA        262     A
CGIA     *         EQU"S FOR EQUVIALENCED OPCODES                                    CGIA        263     A
CGIA                                                                                  CGIA        264     A
CGIA               MACRO  EQVOC,A,B                                                  CGIA        265     A
CGIA      OC.A     EQU    OC.B                                                       CGIA        266     A
CGIA               RMT                                                               CGIA        267     A
CGIA      .B       =      .A                                                         CGIA        268     A
CGIA               RMT                                                               CGIA        269     A
CGIA               ENDM                                                              CGIA        270     A
CGIA                                                                                  CGIA        271     A
CGIA      END      EQVOC  EOQ         END OF SUBPROGRAM                              CGIA        272     A
CGIA      EOB      EQVOC  EOS         END OF INPUT BUFFER MARKER                     CGIA        273     A
CGIA      EOM      EQVOC  DAR         END OF MACRO EXPANSION                         CGIA        274     A
CGIA                                                                                  CGIA        275     A
CGIA     *         LOCAL BLOCK NAME TABLE                                            CGIA        276     A
CGIA                                                                                  CGIA        277     A
CGIA      LBNT     BSS    0                                                          CGIA        278     A
CGIA               ECHO   1,Z=("HC.FLB")                                             CGIA        279     A
CGIA               DATA   10H_Z                                                      CGIA        280     A
CGIA      N.LB     EQU    *-LBNT      NUMBER OF FIXED LOCAL BLOCKS                   CCGA072       7     A
CGIA      .XL      IFNE   HC.NXLB,0                                                  CCGA072       8     A
CGIA      .X       SET    1                                                          CCGA072       9     A
CGIA               DUP    HC.NXLB                                                    CCGA072      10     A
CGIA      .1       DECMIC .X,3                                                       CCGA072      11     A
CGIA      .X       SET    .X+1                                                       CCGA072      12     A
CGIA               DATA   10HLBLK".1"                                                CCGA072      13     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA              ENDD                                                          CCGA072      14     A
CGIA      .XL     ENDIF                                                         CCGA072      15     A
CGIA      LTCODES SPACE  3,15                                                   CGIA        281     A
CGIA     **       LOADER TABLE CODES                                            CGIA        282     A
CGIA                                                                            CGIA        283     A
CGIA      LT.IDT  =      7700B       IDENT OR PREFIX TABLE                      CGIA        284     A
CGIA      LT.LDSET =     7000B       LDSET                                      CGIA        285     A
CGIA      LT.PIDL =      3400B       PROGRAM ID AND LENGTH                      CGIA        286     A
CGIA      LT.ENTR =      3600B       ENTRY POINTS                               CGIA        287     A
CGIA      LT.XTXT =      3700B       XTEXT ( USED INSTEAD OF TEXT )             CGIA        288     A
CGIA      LT.REPI =      4700B       XREPL                                      CGIA        289     A
CGIA      LT.VFDP =      3500B       VFDP                                       CGIA        290     A
CGIA      LT.XFIL =      4100B       XFILL                                      CGIA        291     A
CGIA      LT.FILL =      4200B                                                  CGIA        292     A
CGIA      LT.LINK =      4400B       LINK                                       CGIA        293     A
CGIA      LT.XLNK =      4500B       XLINK                                      CGIA        294     A
CGIA      LT.LNT  =      5700B       LINE NUMBER TABLE FOR *FID*                CGIA        295     A
CGIA      LT.XFER =      4600B       TRANSFER NAME                              CGIA        296     A
CGIA                                                                            CGIA        297     A
CGIA      LS.LIB  =      10B         LDSET LIB=                                 CGIA        298     A
CGIA      LS.USE  =      16B         LDSET USE=                                 CGIA        299     A
CGIA      IDT     SPACE  2,14                                                   CGIA        300     A
CGIA     **       IDENT  TABLE                                                  CGIA        301     A
CGIA                                                                            CGIA        302     A
CGIA      TMVM    MICRO  2,1,/"MDL"/    71   72   73   74   175  176           CGIA        303     A
CGIA      TMVM    MICRO  5*"TMVM"-1,5,/6464 6464 6464 666X C5CX 767X /          CGIA        304         I
         -CCG0001
CGIA      TMVM    MICRO  5*"TMVM"-4,5,/6464 6464 6464 666X C5CX 767X /          CCG0001       5     A
CGIA                                                                            CGIA        305     A
CGIA      IDT     BSS    0                                                      CGIA        306     A
CGIA              LOC    0                                                      CGIA        307     A
CGIA              VFD    12/LT.IDT,12/16B,36/0                                  CGIA        308     A
CGIA              DATA   7LPROGNAM                                              CGIA        309     A
CGIA              DATA   20H MM/DD/YY  HH.MM.SS   DATE AND TIME                 CGIA        310     A
CGIA              DATA   10H"OS.ID"                                             CGIA        311     A
CGIA              DATA   10H"LPNAME""VER"                                       CGIA        312     A
CGIA              VFD    30/5H"MODLVL",30/5L"TMVM"                              CGIA        313     A
CGIA              DATA   10H I            TYPE = RELOC, DEPENDENCIES = INT MULT CGIA        314     A
CGIA      CCO     BSSZ   3           CC OPTIONS                                 CGIA        315     A
CGIA      COMMNT  BSSZ   4           COMMENTS                                   CGIA        316     A
CGIA              LOC    *O                                                     CGIA        317     A
CGIA      TT      SPACE  2,12                                                   CGIA        318     A
CGIA     *        TEXT TABLE                                                    CGIA        319     A
CGIA                                                                            CGIA        320     A
CGIA      TI      OBSS   1           TEXT INDEX ( 0 - 15 )                      CGIA        321     A
CGIA      TH      OBSS   1           12/LT.XTXT,12/WC,12/RB,24/ORGC             CGIA        322     A
CGIA      TR      OBSS   1           TEXT RELOCATION BYTES                      CGIA        323     A
CGIA      TD      OBSS   15          TEXT  DATA                                 CGIA        324     A
CGIA                                                                            CGIA        325     A
CGIA      AI      VFD    12/2003B,48/0     ADDR IN CURRENT BLOCK ( AD. FORMAT ) CGIA        326     A
CGIA      CW      BSS    1           CURRENT TEXT WORD                          CGIA        327     A
CGIA                                                                            CGIA        328     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA      RBWORD   OBSS   3           ADDRESS RELOC INFO IN *AR.* FORMAT                CGIA      329    A
CGIA      IH       =      RBWORD+1    SYMTAB ORDINAL OF SYMBOL                          CGIA      330    A
CGIA     * IH+1                       0L_EXT-NAME IF RL = 3                             CGIA      331    A
CGIA      WB       OBSS   1           WORD B OF SYMTAB ENTRY                            CGIA      332    A
CGIA      IH2      BSSZ   2           SYMTAB ORD OF H2 AND TRUC *CA* FOR *FMI*          CGIA      333    A
CGIA      GNIW     TITLE  BASIC I/O MACROS AND ROUTINES                                CGIA      334    A
CGIA     **        GNIW - GET NEXT INPUT WORD FROM *IBUF*                              CGIA      335    A
CGIA                                                                                   CGIA      336    A
CGIA      GNIW     MACRO                                                               CGIA      337    A
CGIA               SX7    A5-EOBA                                                      CGIA      338    A
CGIA     +         NZ     X7,*+1                                                       CGIA      339    A
CGIA               RJ     FIB                                                          CGIA      340    A
CGIA               SA5    A5+B1                                                        CGIA      341    A
CGIA               ENDM                                                                CGIA      342    A
CGIA                                                                                   CGIA      343    A
CGIA      FIB      ROUTINE                                                             CGIA      344    A
CGIA               READW  F.SLST,IBUF,L.IBUF                                           CGIA      345    A
CGIA               SA5    IBUF-1                                                       CGIA      346    A
CGIA               EQ     FIB                                                          CGIA      347    A
CGIA      WRTEXT   SPACE  3,8                                                          CGIA      348    A
CGIA     **        WRTEXT - OUTPUT DATA TO A TEXT TABLE                                CGIA      349    A
CGIA     *                                                                             CGIA      350    A
CGIA     *         WRTEXT NP,RELOC    AND  (X4) = DATA RIGHT ADJUSTED                  CGIA      351    A
CGIA     *                NP = NUMBER OF PARCELS OF DATA ( 0 - 4  )                   CGIA      352    A
CGIA     *                RELOC - PRESENT ONLY IF NP = 4 AND LOWER RELOC SHOULD       CGIA      353    A
CGIA     *                BE ADDED. RELOCATION IS ADDED AUTOMATICALLY FOR NP = 2.     CGIA      354    A
CGIA                                                                                   CGIA      355    A
CGIA      WRTEXT   MACRO  NP,REL                                                       CGIA      356    A
CGIA               R=     B3,NP                                                        CGIA      357    A
CGIA               IFEQ   NP,0,1                                                       CGIA      358    A
CGIA               MX4    0                                                            CGIA      359    A
CGIA               RJ     APT                                                          CGIA      360    A
CGIA               IFC    NE,/REL//,2                                                  CGIA      361    A
CGIA               SA1    RBWORD                                                       CGIA      362    A
CGIA               RJ     ATR                                                          CGIA      363    A
CGIA               ENDM                                                                CGIA      364    A
CGIA      WRLGO    SPACE  3,8                                                          CGIA      365    A
CGIA     **        WRLGO - WRITE WORDS TO LGO FILE IF CO.B " 0                         CGIA      366    A
CGIA     *         ALL WRITES TO THE *LGO* FILE MUST USE THIS MACRO.                   CGIA      367    A
CGIA     *                                                                             CGIA      368    A
CGIA     *         WRLGO  FWA,N.WORDS                                                  CGIA      369    A
CGIA                                                                                   CGIA      370    A
CGIA      WRLGO    MACRO  F,L                                                          CGIA      371    A
CGIA               R=     B6,F                                                         CGIA      372    A
CGIA               R=     B7,L                                                         CGIA      373    A
CGIA               RJ     WWL                                                          CGIA      374    A
CGIA               ENDM                                                                CGIA      375    A
CGIA      WWL      SPACE  2                                                            CGIA      376    A
CGIA      WWL      BSSZ   1           =0 IF BINARY NOT YET WRITTEN                     CGIA      377    A
CGIA               IF     DEF,/DEBUG/LGO,1                                             CGIA      378    A
CGIA               SNAP   *B6,,*B7,NOREGS                                              CGIA      379    A
CGIA      WWL.P    WRITEW F.LGO,B6,B7                                                  CGIA      380    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA      WWL1     EQ     WWL                                                    CGIA      381     A
CGIA      SYM      SPACE  3,14                                                   CGIA      382         I
         -CCG16
CGIA      .T       IFNE   TEST,0                                                 CGIA      383         I
         -CCG16
CGIA                                                                             CGIA      384         I
         -CCG16
CGIA     **        SYM - RETURN DISPLAY CODED VALUE OF SYMBOL NAME IN X6         CGIA      385         I
         -CCG16
CGIA     *                FOR FORTRAN CODED DEBUGGING ROUTINES                  CGIA      386         I
         -CCG16
CGIA     *         CALL                                                          CGIA      387         I
         -CCG16
CGIA     *                NAM = SYM( IH )                                       CGIA      388         I
         -CCG16
CGIA                                                                             CGIA      389         I
         -CCG16
CGIA      SYM1     SB1    1                                                      CGIA      390         I
         -CCG16
CGIA               RJ     CSN         CONVERT ORDINAL TO BCD NAME                CGIA      391         I
         -CCG16
CGIA               MX5    48                                                     CGIA      392         I
         -CCG16
CGIA               BX1    X5*X6                                                  CGIA      393         I
         -CCG16
CGIA               RJ     FSN                                                    CGIA      394         I
         -CCG16
CGIA               SA3    =1H                                                    CGIA      395         I
         -CCG16
CGIA               SB2    B3-B1                                                  CGIA      396         I
         -CCG16
CGIA               MX2    B2                                                     CGIA      397         I
         -CCG16
CGIA               BX3    -X2*X3                                                 CGIA      398         I
         -CCG16
CGIA               BX6    X1+X3       SPACE FILL NAME                            CGIA      399         I
         -CCG16
CGIA      SYM      ENTRY.                                                        CGIA      400         I
         -CCG16
CGIA               SA2    X1+                                                    CGIA      401         I
         -CCG16
CGIA               NZ     X2,SYM1                                                CGIA      402         I
         -CCG16
CGIA               SA3    =1H                                                    CGIA      403         I
         -CCG16
CGIA               BX6    X3                                                     CGIA      404         I
         -CCG16
CGIA               EQ     SYM                                                    CGIA      405         I
         -CCG16
CGIA                                                                             CGIA      406         I
         -CCG16
CGIA     **        GETNSYM - RETURN N.SYMS IN SYMBOL TABLE                       CGIA      407         I

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
                -CCG16
CGIA                                                                    CGIA        408         I
                -CCG16
CGIA             GETNSYM  ENTRY.                                        CGIA        409         I
                -CCG16
CGIA                      SA2     =XL.SYM                               CGIA        410         I
                -CCG16
CGIA                      SX3     3                                     CGIA        411         I
                -CCG16
CGIA                      IX6     X2/X3,B2                              CGIA        412         I
                -CCG16
CGIA                      SA6     X1                                    CGIA        413         I
                -CCG16
CGIA                      EQ      GETNSYM                               CGIA        414         I
                -CCG16
CGIA                                                                    CGIA        415         I
                -CCG16
CGIA             .T       ENDIF                                         CGIA        416         I
                -CCG16
CGIA             ADW      TITLE   TS MODE TABLE MANAGER INTERFACE       CGIA        417         I
                -CCGA005
CGIA             SETXTH   SPACE   3,14                                  CCGA005      10     A
CGIA            **        SETXTH - SETUP XTEXT HEADER WORD              CCGA005      11     A
CGIA            *         CALL -                                        CCGA005      12     A
CGIA            *         SETXTH AIW,THWA                               CCGA005      13     A
CGIA            *                AIW = X-REGISTER THAT *AI* IS IN       CCGA005      14     A
CGIA            *                THWA = A-REGISTER WITH ADDRESS OF *TH* OR BLANK  CCGA005  15  A
CGIA                                                                    CCGA005      16     A
CGIA             SETXTH   MACRO   AIW,THWA                              CCGA005      17     A
CGIA                      SB4     LT.XTXT-2000B                         CCGA005      18     A
CGIA                      MX7     -AD.CRBOL                             CCGA005      19     A
CGIA                      BX7     -X7*AIW                               CCGA005      20     A
CGIA                      PX6     B4,X7                                 CCGA005      21     A
CGIA                      SA6     THWA TH                               CCGA005      22     A
CGIA                      ENDM                                          CCGA005      23     A
CGIA             TSMODE   SPACE   4,14                                  CCGA005      24     A
CGIA            **        TS MODE TABLE MANAGER INTERFACE               CCGA005      25     A
CGIA             .TS      IFNE    HC.TS,0                               CGIA        418     A
CGIA                                                                    CGIA        419     A
CGIA            **        ADW - ADD 1 WORD TO THE END OF A MANAGED TABLE CGIA       420     A
CGIA                                                                    CGIA        421     A
CGIA             ADW1     SA6     ADWA                                  CGIA        422     A
CGIA                      RJ      ATS=                                  CGIA        423     A
CGIA                      SA3     ADWA                                  CGIA        424     A
CGIA                      BX6     X3                                    CGIA        425     A
CGIA                      SA6     B7-B1                                 CGIA        426     A
CGIA             ADW=     ROUTINE                                       CGIA        427     A
CGIA                      BX6     X1                                    CGIA        428     A
CGIA                      SX1     B1                                    CGIA        429     A
CGIA                      EQ      ADW1                                  CGIA        430     A
CGIA                                                                    CGIA        431     A
CGIA             ADWA     BSS     1                                     CGIA        432     A

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA      ATS      SPACE  3                                                    CGIA        433     A
CGIA     **        ATS= - ALLOCATE TABLE SPACE                                 CGIA        434     A
CGIA                                                                           CGIA        435     A
CGIA      ATS1     SA1    =XFTAB=+A0                                           CGIA        436     A
CGIA               RJ     =XALC=      CALL TS TABLE MANAGER                    CGIA        437     A
CGIA               SA3    A2                                                   CGIA        438     A
CGIA               BX0    X4                                                   CGIA        439     A
CGIA               SA2    A1+                                                  CGIA        440     A
CGIA      ATS=     ROUTINE                                                     CGIA        441     A
CGIA               BX4    X0                                                   CGIA        442     A
CGIA               LX0    X1                                                   CGIA        443     A
CGIA               EQ     ATS1                                                 CGIA        444     A
CGIA                                                                           CGIA        445     A
CGIA      .TS      ENDIF                                                       CGIA        446     A
CGIA      CGIA     TITLE  MAIN LOOP AND MACRO REFERENCE PROCESSING             CGIA        447     A
CGIA     **        CGIA - CODE GENERATOR INTERNAL ASSEMBLER                    CGIA        448     A
CGIA                                                                           CGIA        449     A
CGIA      CG$IA    ENTRY. **                                                   CGIA        450     A
CGIA               SX7    IA.EX                                                CGIA        451     A
CGIA               SA7    =XCG.MOX#   CG.MOX = IA.EX                           CGIA        452     A
CGIA               MX6    0                                                    CGIA        453     A
CGIA               SA6    L.LIB       L.LIB = 0                                CGIA        454     A
CGIA               SA6    L.LAFT      L.LAFT = 0                               CCG14         1     A
CGIA               ALLOC  LIB,2       ALLOC( LIB , 2 )  */ SPACE FOR HEADER WDS CGIA        455     A
CGIA               SA1    =XHO$C                                               CGIA        456     A
CGIA               SA2    =XHO$LO$O                                            CGIA        457     A
CGIA               MX6    0           BCDC = 0                                 CGIA        458     A
CGIA               SA6    =XHO$CSN    HO$CSN = 0                               CGIA        459     A
CGIA               ZR     X1,CGIA1    IF C = 0                                 CGIA        460     A
CGIA               SA6    HO$B        HO$B = 0      */ IN CASE HOST HASNT DONE IT CGIA     461     A
CGIA               MX6    1           BCDC = 1S59                              CGIA        462     A
CGIA               EQ     CGIA2                                                CGIA        463     A
CGIA      CGIA1    ZR     X2,CGIA2    IF OL = 0                                CGIA        464     A
CGIA               SX6    1           BCDC = 1                                 CGIA        465     A
CGIA                                                                           CGIA        466     A
CGIA      CGIA2    SA6    BCDC                                                 CGIA        467     A
CGIA               SA1    =XHO$B                                               CGIA        468     A
CGIA               NZ     X6,CGIA3    IF BCDC " 0   */ OL ! C SELECTED         CGIA        469     A
CGIA               SA3    APTA+1                                               CGIA        470     A
CGIA               BX7    X3          SET FOR NO OBJECT LIST                   CGIA        471     A
CGIA               SA7    APT5                                                 CGIA        472     A
CGIA               ZR     X1,CG$IA    IF B = 0                                 CGIA        473     A
CGIA      CGIA3    NZ     X1,CGIA4    IF B " 0                                 CGIA        474     A
CGIA               SA2    WWL1                                                 CGIA        475     A
CGIA               BX7    X2          PLUG LGO WRITE ROUTINE                   CGIA        476     A
CGIA               SA7    WWL.P                                                CGIA        477     A
CGIA                                                                           CGIA        478     A
CGIA     *         FLUSH, REWIND AND BEGIN READING THE *SLIST* FILE            CGIA        479     A
CGIA                                                                           CGIA        480     A
CGIA      CGIA4    WRITER F.SLST,,R                                            CGIA        481     A
CGIA               REWIND F.SLST                                               CGIA        482     A
CGIA               READ   F.SLST                                               CGIA        483     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA              SX6    1S10+OC.EOB                                          CGIA        484     A
CGIA              LX6    48                                                   CGIA        485     A
CGIA              SA6    IBUFL       SETUP END OF BUFFER SENTINAL             CGIA        486     A
CGIA                                                                          CGIA        487     A
CGIA     **       EOB - END OF INPUT BUFFER, READ NEXT N WORDS INTO WORKING BUF CGIA      488     A
CGIA                                                                          CGIA        489     A
CGIA              PROCESS EOB                                                  CGIA        490     A
CGIA              READW  F.SLST,IBUF,L.IBUF                                    CGIA        491     A
CGIA              MI     X1,*+1S17         IF EOF ON INPUT                     CGIA        492     A
CGIA              SA2    L.LAFT                                               CCG14         2     A
CGIA              SA5    IBUF-1      II = IBUF - 1                            CGIA        493     A
CGIA              SX6    X2-7700B                                             CCG14         3     A
CGIA              MI     X6,GNIW     IF L.LAFT < 7700B  */ NO OVERFLOW        CCG14         4     A
CGIA              RJ     DAT         DUMP THE LINK AND FILL TABLES            CCG14         5     A
CGIA                                                                          CGIA        494     A
CGIA     **       GNIW - GET NEXT INPUT WORD ( MAIN LOOP )                     CGIA        495     A
CGIA                                                                          CGIA        496     A
CGIA      GNIW    SA5    A5+1        II = II + 1                              CGIA        497     A
CGIA              SX7    0                                                    CGIA        498     A
CGIA              MX0    -6                                                   CGIA        499     A
CGIA              UX6    B2,X5       OPC = OC[II]                             CGIA        500     A
CGIA              SA7    IH          IH = 0                                   CGIA        501     A
CGIA              PL     X5,GNIW1    IF [II] > 0                              CGIA        502     A
CGIA              SB3    B2-128                                               CGIA        503     A
CGIA              GE     B3,PMR      IF OPC \ 128  */ A MACRO REF             CGIA        504     A
CGIA                                                                          CGIA        505     A
CGIA     **       ENTER OPCODE PROCESSOR WITH -                                CGIA        506     A
CGIA     *        (X0) = MASK(54) , (A5,X5) = II, [II]                         CGIA        507     A
CGIA     *        (B2) = OPCODE, (X6) = LOW48[II] AND (X4) = )OPC              CGIA        508     A
CGIA                                                                          CGIA        509     A
CGIA      GNIW1   SA4    OC.AT+B2                                             CGIA        510     A
CGIA              SB3    X4                                                   CGIA        511     A
CGIA              AX4    36                                                   CGIA        512     A
CGIA              JP     B3          JUMP( OC.AT[OC] )                        CGIA        513     A
CGIA      IA.EX   SPACE  3,14                                                 CGIA        514     A
CGIA     **       IA.EX - ERROR EXIT PROCESSING                                CGIA        515     A
CGIA     *        CALLED BY THE TABLE MANAGER WHEN A MEMORY OVERFLOW           CGIA        516     A
CGIA     *        OR AT THE END OF THE SLIST FILE IF THERE ARE ASSEMBLY ERRORS CGIA        517     A
CGIA                                                                          CGIA        518     A
CGIA      IA.EX   SA1    HO$B                                                 CGIA        519     A
CGIA              ZR     X1,IA.EX2   IF B = 0                                 CGIA        520     A
CGIA                                                                          CGIA        521     A
CGIA              SA2    WWL                                                  CGIA        522     A
CGIA              ZR     X2,IA.EX1   IF WWL = 0    */ NOTHING ON LGO          CGIA        523     A
CGIA                                                                          CGIA        524     A
CGIA              WRITER F.LGO                                                CGIA        525     A
CGIA              BKSP   F.LGO,,R                                             CGIA        526     A
CGIA                                                                          CGIA        527     A
CGIA      IA.EX1  WRLGO  IDT,17B     OUTPUT IDENT TABLE                       CGIA        528     A
CGIA              WRLGO  ERRL,3      AND ERROR LINE                           CGIA        529     A
CGIA              EQ     CG$IA                                                CGIA        530         I
         -CCG18

         0        1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                EQ     END5              */ GO FLUSH OUT *LGO*                    CCG18          3     A
CGIA                                                                                  CGIA         531     A
CGIA      IA.EX2    SA1    HO$C                                                       CGIA         532     A
CGIA                ZR     X1,CG$IA    IF C = 0                                       CGIA         533     A
CGIA                WRITEC F.CMPS,ERRL,3     OUTPUT ERR LINE                           CGIA         534     A
CGIA                WRITEC F.CMPS,(=8L  END  ),1                                      CGIA         535     A
CGIA                WRITER F.CMPS,,R                                                  CGIA         536     A
CGIA                EQ     CG$IA                                                      CGIA         537     A
CGIA                                                                                  CGIA         538     A
CGIA      ERRL      DATA   28L  ERR   ERRORS IN - PROGNAM                             CGIA         539     A
CGIA      PMR       EJECT                                                             CGIA         540     A
CGIA     **         PMR - PROCESS MACRO REFERENCE                                     CGIA         541     A
CGIA                                                                                  CGIA         542     A
CGIA      PMR       SA4    =XF$MACS+B3 MIW = MACS(OPC)                                CGIA         543     A
CGIA                SA3    X4                                                         CGIA         544     A
CGIA                BX6    X4                                                         CGIA         545     A
CGIA                SA6    MIW                                                        CGIA         546     A
CGIA                MX7    -3                                                         CGIA         547     A
CGIA                BX6    X3                                                         CGIA         548     A
CGIA                SA6    A6+B1       MIW(2) = MHW                                   CGIA         549     A
CGIA                BX2    -X7*X3      NP = MHW & 7                                   CGIA         550     A
CGIA                SB2    X2                                                         CGIA         551     A
CGIA                                                                                  CGIA         552     A
CGIA     *          UNPACK PARAMETERS FROM MACRO CALL INTO *PBUF*                     CGIA         553     A
CGIA                                                                                  CGIA         554     A
CGIA                MX0    -L.MP                                                      CGIA         555     A
CGIA                BX7    -X0*X5                                                     CGIA         556     A
CGIA                SA7    PBUF+1                                                     CGIA         557     A
CGIA                AX5    L.MP                                                       CGIA         558     A
CGIA                BX7    -X0*X5                                                     CGIA         559     A
CGIA                SA7    A7+B1                                                      CGIA         560     A
CGIA                SB3    B1+B1                                                      CGIA         561     A
CGIA                LE     B2,B3,PMR1  IF NP @ 2                                      CGIA         562     A
CGIA                GNIW                                                              CGIA         563     A
CGIA                BX7    -X0*X5                                                     CGIA         564     A
CGIA                SA7    PBUF+3                                                     CGIA         565     A
CGIA                AX5    L.MP                                                       CGIA         566     A
CGIA                BX7    -X0*X5                                                     CGIA         567     A
CGIA                SA7    A7+B1                                                      CGIA         568     A
CGIA                SA4    MIW+1                                                      CGIA         569     A
CGIA                MX3    -3                                                         CGIA         570     A
CGIA                BX2    -X3*X4                                                     CGIA         571     A
CGIA                SX6    X2-5                                                       CGIA         572     A
CGIA                MI     X6,PMR1     IF NP < 5                                      CGIA         573     A
CGIA                GNIW                                                              CGIA         574     A
CGIA                BX7    -X0*X5                                                     CGIA         575     A
CGIA                SA7    PBUF+5                                                     CGIA         576     A
CGIA                AX5    L.MP                                                       CGIA         577     A
CGIA                BX7    -X0*X5                                                     CGIA         578     A
CGIA                SA7    A7+B1                                                      CGIA         579     A
CGIA                                                                                  CGIA         580     A
CGIA      PMR1      SA1    BCDC                                                       CGIA         581     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA              MI     X1,PMR5     IF BCDC < 0   */ C OPTION ONLY                CGIA      582     A
CGIA                                                                              CGIA      583     A
CGIA     *        MOVE MACRO TEXT TO MACRO BUFFER                                 CGIA      584     A
CGIA                                                                              CGIA      585     A
CGIA              SX7    A5                                                        CGIA      586     A
CGIA              SB2    OC.EOM                                                    CGIA      587     A
CGIA              PX6    B2,X7                                                     CGIA      588     A
CGIA              SA5    MIW         6/,18/N.PR,18/LEN,18/FWA-1                    CGIA      589     A
CGIA              SX2    X5+B1                                                     CGIA      590     A
CGIA              AX5    18                                                        CGIA      591     A
CGIA              SX1    X5                                                        CGIA      592     A
CGIA              IX0    X2+X1       PRA = FWA + LEN                               CGIA      593     A
CGIA              LX1    18                                                        CGIA      594     A
CGIA              BX6    X1+X6                                                     CGIA      595     A
CGIA              LX1    -18                                                       CGIA      596     A
CGIA              SA6    =XF$MXB+X1  MXB(LEN) = PACK(OC.EOM,II)                    CGIA      597     A
CGIA              ZR     X1,PMR3     IF LEN = 0                                    CGIA      598     A
CGIA              MOVE   X1,X2,F$MXB MOVE( LEN, FWA , F$MXB )                      CGIA      599     A
CGIA      PMR     SPACE  1,4                                                       CGIA      600     A
CGIA     *        SUBSTITUTE PARAMETER VALUES IN THE MACRO BUFFER                 CGIA      601     A
CGIA     *        RELOC BYTE FORMAT IS 6/LOW-BIT,6/WORD INDEX                     CGIA      602     A
CGIA                                                                              CGIA      603     A
CGIA              SA4    X0          PRW = [PRA]                                   CGIA      604     A
CGIA              MX7    -IH.CAL                                                   CGIA      605     A
CGIA              SB2    60-L.MRB    SC = 60 - L.MRB                               CGIA      606     A
CGIA              MX0    -6                                                        CGIA      607     A
CGIA              AX5    18                                                        CGIA      608     A
CGIA              SB4    X5          N = N.PR[MIW] */ N. SUBSTIT PARAMS            CGIA      609     A
CGIA              SB5    F$MXB       BASE = F$MXB                                  CGIA      610     A
CGIA                                                                              CGIA      611     A
CGIA      PMR2    ZR     B4,PMR3     IF N = 0      */ END OF RELOC BYTES           CGIA      612     A
CGIA              AX3    B2,X4       RB = SHIFT( PRW , -SC )                       CGIA      613     A
CGIA              BX6    -X0*X3                                                    CGIA      614     A
CGIA              SA2    B5+X6       MW = BASE( WI[RB] )                           CGIA      615     A
CGIA              SB5    B5+X6       BASE = BASE + WI[RB]                          CGIA      616     A
CGIA              AX3    6                                                         CGIA      617     A
CGIA              BX3    -X0*X3                                                    CGIA      618     A
CGIA              SB3    X3-60                                                     CGIA      619     A
CGIA              LX2    -B3                                                       CGIA      620     A
CGIA              SA1    X2                                                        CGIA      621     A
CGIA              BX2    X7*X2                                                     CGIA      622     A
CGIA              SB3    X3                                                        CGIA      623     A
CGIA              BX6    X2+X1       SUBSTITUTE PARAM VALUE IN MACRO WORD          CGIA      624     A
CGIA              SB4    B4-B1       N = N - 1                                     CGIA      625     A
CGIA              LX6    B3                                                        CGIA      626     A
CGIA              SB2    B2-L.MRB    SC = SC - L.MRB                               CGIA      627     A
CGIA              SA6    A2                                                        CGIA      628     A
CGIA              GE     B2,PMR2     IF SC \ 0                                     CGIA      629     A
CGIA              SB2    B2+60       SC = SC + 60                                  CGIA      630     A
CGIA              SA4    A4+B1       PRA = PRA + 1;  PRW = [PRA]                   CGIA      631     A
CGIA              EQ     PMR2                                                      CGIA      632     A
CGIA                                                                              CGIA      633     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA      PMR3     SA1    BCDC                                                    CGIA      634    A
CGIA               ZR     X1,PMR4     IF BCDC = 0   */ NO OBJ LIST                CGIA      635    A
CGIA                                                                              CGIA      636    A
CGIA               SA1    MIW+1                                                   CGIA      637    A
CGIA               RJ     FMC         FORMAT AND STORE MACRO CALL LINE            CGIA      638    A
CGIA                                                                              CGIA      639    A
CGIA      PMR4     SA5    F$MXB-1     II = F$MXB - 1                              CGIA      640    A
CGIA               EQ     GNMW1                                                   CGIA      641    A
CGIA                                                                              CGIA      642    A
CGIA      PMR5     SA1    MIW+1                                                   CGIA      643    A
CGIA               RJ     FMC         FORMAT AND OUTPUT MACRO LINE                CGIA      644    A
CGIA                                                                              CGIA      645    A
CGIA      PMR6     SX6    0                                                       CGIA      646    A
CGIA               SA6    MIW         MIW = 0       */ CLEAR MACRO MODE FLAG      CGIA      647    A
CGIA               EQ     GNIW        EXIT TO MAIN LOOP                           CGIA      648    A
CGIA      EOM      SPACE  2                                                       CGIA      649    A
CGIA     **        EOM - END OF MACRO PROCESSING                                 CGIA      650    A
CGIA                                                                              CGIA      651    A
CGIA               PROCESS EOM                                                   CGIA      652    A
CGIA               SA5    X5          II = [II]     */ RESTORE INPUT BUFFER POINT CGIA      653    A
CGIA               AX6    18                                                      CGIA      654    A
CGIA               NZ     X6,PMR6     IF LENGTH " 0                               CGIA      655    A
CGIA               WRTEXT 0           LIST THE MACRO LINE                         CGIA      656    A
CGIA               EQ     PMR6                                                    CGIA      657    A
CGIA      INST     TITLE  MACHINE INSTRUCTION PROCESSORS                          CGIA      658    A
CGIA     **        MACHINE CODE INSTRUCTION PROCESSORS                           CGIA      659    A
CGIA     *         CONVERT *SI.* FORMAT TO MACHINE INSTRUCTIONS, NOTE RELOCATION CGIA      660    A
CGIA     *         WRITE TO TEXT TABLES, CONVERT TO BCD, ETC.                    CGIA      661    A
CGIA      PROCESS  SPACE  2                                                       CGIA      662    A
CGIA     **        PROCESS - DEFINE TRANSFER POINT FOR OPCODE PROCESSORS         CGIA      663    A
CGIA     *                                                                       CGIA      664    A
CGIA     *         PROCESS (PLIST),V                                             CGIA      665    A
CGIA     *         PLIST - LIST OF 1 OR MORE OPCODES TO BE PROCESSED             CGIA      666    A
CGIA     *         V = COMMON VALUE ASSOCIATED WITH THE LIST OF NAMES            CGIA      667    A
CGIA                                                                              CGIA      668    A
CGIA               PURGMAC PROCESS                                               CGIA      669    A
CGIA      PROCESS  MACRO  P,V                                                     CGIA      670    A
CGIA               IRP    P                                                       CGIA      671    A
CGIA               NOREF  .P                                                      CGIA      672    A
CGIA      .P       BSS    0                                                       CGIA      673    A
CGIA               IFC    NE,/V//,2                                               CGIA      674    A
CGIA               NOREF  )P                                                      CGIA      675    A
CGIA      )P       =      V                                                       CGIA      676    A
CGIA               IRP                                                            CGIA      677    A
CGIA               ENDM                                                           CGIA      678    A
CGIA      PRTII    SPACE  2,10                                                    CGIA      679    A
CGIA     **        PRTII - PROCESS TYPE I INSTRUCTION                            CGIA      680    A
CGIA     *                                                                       CGIA      681    A
CGIA     *         PRTII  (PLIST)                                                CGIA      682    A
CGIA                                                                              CGIA      683    A
CGIA      PRTII    MACRO  P                                                       CGIA      684    A
CGIA               IRP    P                                                       CGIA      685    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA              NOREF  .P,)P                                                CGIA     686    A
CGIA      )P      =      OC.P*1S9                                             CGIA     687    A
CGIA      .P      BSS    0                                                    CGIA     688    A
CGIA              ENDM                                                        CGIA     689    A
CGIA      PSEUDO  SPACE  3,12                                                 CGIA     690    A
CGIA     **       PSEUDO - DEFINE PSEUDO OP NAME AND ARGUMENT TYPES FOR BCD   CGIA     691    A
CGIA     *        LISTING PURPOSES                                            CGIA     692    A
CGIA     *                                                                    CGIA     693    A
CGIA     *        PSEUDO NAME,(ARG-TYPES),LAB                                 CGIA     694    A
CGIA     *        LAB = 1 IF FIRST ARG IS IN THE LABEL FIELD, ELSE 0          CGIA     695    A
CGIA                                                                          CGIA     696    A
CGIA      PSEUDO  MACRO  NAME,PL,LAB                                          CGIA     697    A
CGIA     +        VFD    36/6L;A                                              CGIA     698    A
CGIA              VFD    3/1-LAB                                              CGIA     699    A
CGIA      .N      SET    0                                                    CGIA     700    A
CGIA              IRP    PL                                                   CGIA     701    A
CGIA      .N      SET    .N+1                                                 CGIA     702    A
CGIA              VFD    3/AT.PL                                              CGIA     703    A
CGIA              IRP                                                         CGIA     704    A
CGIA              VFD    *P/.N                                                CGIA     705    A
CGIA              ENDM                                                        CGIA     706    A
CGIA                                                                          CGIA     707    A
CGIA              ECHO   1,Z=(C,B,S,Q,M,E),V=(1,2,3,4,5,6)                    CGIA     708    A
CGIA      AT.Z    =      V                                                    CGIA     709    A
CGIA              NOREF  .N                                                   CGIA     710    A
CGIA      XBX     SPACE  2,8                                                  CGIA     711    A
CGIA      .TS     IFEQ   HC.TS,0                                              CGIA     712    A
CGIA                                                                          CGIA     713    A
CGIA     **       XBX INSTRUCTIONS                                            CGIA     714    A
CGIA                                                                          CGIA     715    A
CGIA              PRTII  (ILS,IRS,NR,RNZ,UP,PK)                               CGIA     716    A
CGIA              SX3    200020B                                              CGIA     717    A
CGIA              EQ     TYI                                                  CGIA     718    A
CGIA                                                                          CGIA     719    A
CGIA     **       XXX INSTRUCTIONS                                            CGIA     720    A
CGIA                                                                          CGIA     721    A
CGIA                                                                          CGIA     722    A
CGIA              PRTII  (AND,OR,XOR,FA,FS,DFA,DFS,RFA,RFS,IA,IS,FM,RFM,DFM,FD,_  CGIA     723    A
CGIA     ,RFD)                                                                CGIA     724    A
CGIA              ECHO   1,X=(IAZ,ISZ,IM),Y=(IA,IS,DFM)                       CGIA     725    A
CGIA              PROCESS X,)Y                                                CGIA     726    A
CGIA              SX3    202020B                                              CGIA     727    A
CGIA                                                                          CGIA     728    A
CGIA      TYI     SX6    307070B                                              CGIA     729    A
CGIA              BX7    X6*X5                                                CGIA     730    A
CGIA              BX6    X3-X7                                                CGIA     731    A
CGIA              SA6    RERR                                                 CGIA     732    A
CGIA              ZR     X6,TYI1A    IF RERR = 0                              CGIA     733    A
CGIA                                                                          CGIA     734    A
CGIA              SA1    NAE                                                  CGIA     735    A
CGIA              SX6    X1+B1       NAE = NAE + 1 */ BUMP ERROR COUNTER      CGIA     736    A
CGIA              SA6    A1                                                   CGIA     737    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                                                                                    CGIA        738     A
CGIA      TYI1A    MX0     -3                                                           CGIA        739     A
CGIA               BX1     -X0*X5      I = RI[II] & 7                                   CGIA        740     A
CGIA               LX5     -SI.RJP                                                      CGIA        741     A
CGIA               BX2     -X0*X5      J = RJ[II] & 7                                   CGIA        742     A
CGIA               LX5     SI.RJP-SI.RKP                                                CGIA        743     A
CGIA               BX3     -X0*X5      K = RK[II] & 7                                   CGIA        744     A
CGIA                                                                                    CGIA        745     A
CGIA     *         FORM FMIJK , (X4) = FM000, (X1,X2,X3) = I,J,K                        CGIA        746     A
CGIA                                                                                    CGIA        747     A
CGIA      TYI1     LX1     6                                                            CGIA        748     A
CGIA               BX4     X1+X4                                                        CGIA        749     A
CGIA               LX2     3                                                            CGIA        750     A
CGIA               BX3     X2+X3                                                        CGIA        751     A
CGIA               IX4     X3+X4                                                        CGIA        752     A
CGIA               WRTEXT 1                                                             CGIA        753     A
CGIA               EQ      GNIW                                                         CGIA        754     A
CGIA                                                                                    CGIA        755     A
CGIA     *         BXI  -XJ*XK INSTRUCTIONS, MACHINE FORMAT IS - FMIKJ .                CGIA        756     A
CGIA                                                                                    CGIA        757     A
CGIA               PRTII   (STR,IMP,EQV)                                                CGIA        758     A
CGIA               MX0     -3                                                           CGIA        759     A
CGIA               BX1     -X0*X5      I = RI[II] & 7                                   CGIA        760     A
CGIA               LX5     -SI.RJP                                                      CGIA        761     A
CGIA               BX3     -X0*X5      K = RJ[II] & 7                                   CGIA        762     A
CGIA               LX5     SI.RJP-SI.RKP                                                CGIA        763     A
CGIA               BX2     -X0*X5      J = RK[II] & 7                                   CGIA        764     A
CGIA               EQ      TYI1                                                         CGIA        765     A
CGIA                                                                                    CGIA        766     A
CGIA     *         XX INSTRUCTIONS                                                      CGIA        767     A
CGIA                                                                                    CGIA        768     A
CGIA               PRTII   (XMT,XMTC,CX)                                                CGIA        769     A
CGIA               MX0     -3                                                           CGIA        770     A
CGIA               BX1     -X0*X5      I = RJ[II] & 7                                   CGIA        771     A
CGIA               LX5     -SI.RJP                                                      CGIA        772     A
CGIA               BX2     -X0*X5      J = RJ[II] & 7                                   CGIA        773     A
CGIA               BX3     -X0*X5      K = J                                            CGIA        774     A
CGIA               EQ      TYI1                                                         CGIA        775     A
CGIA                                                                                    CGIA        776     A
CGIA     **        CONSTANT SHIFT AND MASK INSTRUCTIONS                                 CGIA        777     A
CGIA                                                                                    CGIA        778     A
CGIA               PRTII   (KLS,KRS,FMA)                                                CGIA        779     A
CGIA               MX0     -3                                                           CGIA        780     A
CGIA               BX1     -X0*X5      I = RI[II] & 7                                   CGIA        781     A
CGIA               LX5     -SI.CAP                                                      CGIA        782     A
CGIA               MX2     0           J = 0                                            CGIA        783     A
CGIA               SX3     X5          K = CA[II]                                       CGIA        784     A
CGIA               EQ      TYI1                                                         CGIA        785     A
CGIA                                                                                    CGIA        786     A
CGIA      #DAL     IFNE    .DAL,0                                                       CGIA        787     A
CGIA                                                                                    CGIA        788     A
CGIA     **        LCM READ/WRITE INSTRUCTIONS                                          CGIA        789     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                                                                    CGIA        790     A
CGIA                PROCESS DRL,01400B                                  CGIA        791     A
CGIA                PROCESS DWL,01500B                                  CGIA        792     A
CGIA                SX1    B0          I = 0                            CGIA        793     A
CGIA                MX0    -3                                           CGIA        794     A
CGIA                BX2    -X0*X5      J = RI[II]                       CGIA        795     A
CGIA                LX5    -SI.RJP                                      CGIA        796     A
CGIA                BX3    -X0*X5      K = RJ[II]                       CGIA        797     A
CGIA                EQ     TYI1                                         CGIA        798     A
CGIA       #DAL     ENDIF                                               CGIA        799     A
CGIA                                                                    CGIA        800     A
CGIA     **         MEMORY REFERENCE PROCESSING                         CGIA        801     A
CGIA                                                                    CGIA        802     A
CGIA       )F       =      7S8+5S4+6   TO CHANGE BAX INTO 5/6/7          CGIA        803     A
CGIA                                                                    CGIA        804     A
CGIA                PROCESS (SLD,SST,SA),3S8+4S4+6                      CGIA        805     A
CGIA                PROCESS (SDL,SDS,SS),5S4+7                          CGIA        806     A
CGIA                BX1    -X0*X6      IR = RI[II]                      CGIA        807     A
CGIA                LX6    -SI.RJP                                      CGIA        808     A
CGIA                BX2    -X0*X6      JR = RJ[II]                      CGIA        809     A
CGIA                LX6    SI.RJP-SI.RKP                                CGIA        810     A
CGIA                BX3    -X0*X6      KR = RK[II]                      CGIA        811     A
CGIA                BX6    X3                                           CGIA        812     A
CGIA                AX6    3                                            CGIA        813     A
CGIA                BX7    X2                                           CGIA        814     A
CGIA                ZR     X6,SLD1     IF RT[KR] = 0 */ KR IS A B-REGISTER CGIA     815     A
CGIA                BX2    X3                                           CGIA        816     A
CGIA                BX3    X7          SWAP( JR , KR )                  CGIA        817     A
CGIA                                                                    CGIA        818     A
CGIA       SLD1     BX3    -X0*X3                                       CGIA        819     A
CGIA                SB7    B0          FLAG = 0                         CGIA        820     A
CGIA                EQ     PLD2                                         CGIA        821     A
CGIA       PLD      SPACE  2,8                                          CGIA        822     A
CGIA                PROCESS STT,2S8+1                                   CCGA068      11     A
CGIA                SA1    =XS$CON                                      CCGA068      12     A
CGIA                LX5    -SI.IHP                                      CCGA068      13     A
CGIA                BX2    X1-X5                                        CCGA068      14     A
CGIA                LX5    SI.IHP                                       CCGA068      15     A
CGIA                SX1    X2                                           CCGA068      16     A
CGIA                NZ     X1,PLD0     IF IH[SI] " CON.                 CCGA068      17     A
CGIA                                                                    CCGA068      18     A
CGIA                PROCESS LDC,2S8+1                                   CGIA        823     A
CGIA                LX5    -SI.CAP                                      CGIA        824     A
CGIA                SA3    =XO.CUT                                      CGIA        825     A
CGIA                SB3    X5                                           CGIA        826     A
CGIA                SA2    X3+B3       C = CUT( CA[SI] )                CGIA        827     A
CGIA                MX1    -SI.CAL                                      CGIA        828     A
CGIA                BX3    X1*X5                                        CGIA        829     A
CGIA                BX5    X3+X2       CA[SI] = C                       CGIA        830     A
CGIA                LX5    SI.CAP                                       CGIA        831     A
CGIA                                                                    CGIA        832     A
CGIA                PROCESS (LD,ST,STT,PLD,PST,S,ILD,TLD,TST,ILD),2S8+1 CGIA        833         I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

-CCGA068

```
CGIA                PROCESS (LD,ST,PLD,PST,S,ILD,TLD,TST,ILD),2S8+1          CCGA068      19    A
CGIA      PLD0      BSS    0                                                 CCGA068      20    A
CGIA                BX1    -X0*X6      IR = RI[II]                           CGIA        834    A
CGIA                LX6    -SI.RJP                                           CGIA        835    A
CGIA                BX2    -X0*X6      JR = RJ[II]                           CGIA        836    A
CGIA                LX6    SI.RJP-SI.CAIHP                                   CGIA        837    A
CGIA                MX7    -SI.CAIHL                                         CGIA        838    A
CGIA                BX3    -X7*X6      K = CAIH[II]                          CGIA        839    A
CGIA                NZ     X3,PLD1     IF K " 0                              CGIA        840    A
CGIA                SX4    )SLD                                              CGIA        841    A
CGIA                EQ     SLD1                                              CGIA        842    A
CGIA                                                                         CGIA        843    A
CGIA      PLD1      LX5    -SI.CAP                                           CGIA        844    A
CGIA                SB7    B1          FLAG = 1                              CGIA        845    A
CGIA                                                                         CGIA        846    A
CGIA      *         PLD2 - COMMON PROCESSOR TO DETERMINE F AND M FOR A MEMORY REF  CGIA  847    A
CGIA      *         INSTRUCTION.  (B7) = 0 IF EXIT TO TYI1, ELSE FALL THROUGH  CGIA  848    A
CGIA      *         (X1,X2,X3) = IR,JR,KR                                     CGIA        849    A
CGIA      *         EXIT   (X4) = FM000 , (X1,X2) = REDUCED TO REGISTER NUMBERS.  CGIA  850  A
CGIA                                                                         CGIA        851    A
CGIA      PLD2      SB3    3                                                 CGIA        852    A
CGIA                AX6    B3,X2                                             CGIA        853    A
CGIA                LX6    2                                                 CGIA        854    A
CGIA                SB4    X6                                                CGIA        855    A
CGIA                AX4    B4                                                CGIA        856    A
CGIA                MX0    -3                                                CGIA        857    A
CGIA                BX4    -X0*X4      M = SHIFT()XX,-4*(JR/8)) & 7          CGIA        858    A
CGIA                LX4    9                                                 CGIA        859    A
CGIA                                                                         CGIA        860    A
CGIA                AX6    B3,X1                                             CGIA        861    A
CGIA                LX6    2                                                 CGIA        862    A
CGIA                SB4    X6                                                CGIA        863    A
CGIA                SX7    )F                                                CGIA        864    A
CGIA                AX7    B4                                                CGIA        865    A
CGIA                BX6    -X0*X7      F = SHIFT( )F , -4*(IR/8) ) & 7       CGIA        866    A
CGIA                LX6    12                                                CGIA        867    A
CGIA                BX4    X4+X6                                             CGIA        868    A
CGIA                BX1    -X0*X1                                            CGIA        869    A
CGIA                BX2    -X0*X2                                            CGIA        870    A
CGIA                ZR     B7,TYI1     IF FLAG = 0                           CGIA        871    A
CGIA                                                                         CGIA        872    A
CGIA                LX1    6                                                 CGIA        873    A
CGIA                LX2    3                                                 CGIA        874    A
CGIA                BX1    X1+X2                                             CGIA        875    A
CGIA                BX4    X1+X4                                             CGIA        876    A
CGIA                LX4    15          FMIJ0 - 0                             CGIA        877    A
CGIA                MX7    -SI.CAL                                           CGIA        878    A
CGIA                BX3    -X7*X5                                            CGIA        879    A
CGIA                BX4    X3+X4       ADD CA                                CGIA        880    A
CGIA                SX6    X5                                                CGIA        881    A
CGIA                SA6    IH2+1       IH2(2) = CA[II]                       CGIA        882    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA            LX5     SI.CAP-SI.IHP                                                   CGIA        883     A
CGIA            SX3     X5                                                              CGIA        884     A
CGIA            ZR      X3,PLD4     IF IH[II] = 0                                       CGIA        885     A
CGIA                                                                                    CGIA        886     A
CGIA            RJ      GAI         GET ADDR INFO                                       CGIA        887     A
CGIA            MX7     -18                                                             CGIA        888     A
CGIA            SB3     X4                                                              CGIA        889     A
CGIA            SX3     X3+B3       K = K + RA[RBWORD]                                  CGIA        890     A
CGIA            BX4     X7*X4                                                           CGIA        891     A
CGIA            BX3     -X7*X3                                                          CGIA        892     A
CGIA            BX4     X3+X4                                                           CGIA        893     A
CGIA            LX5     SI.IHP+59-SI.H2P                                                CGIA        894     A
CGIA            PL      X5,PLD4     IF ^H2[II]    */ NO H2                              CGIA        895     A
CGIA                                                                                    CGIA        896     A
CGIA            SA3     IH                                                              CGIA        897     A
CGIA            LX3     30                                                              CGIA        898     A
CGIA            BX0     X3+X4       SAVE BINARY AND IH                                  CGIA        899     A
CGIA            GNIW                                                                    CGIA        900     A
CGIA            BX6     X5                                                              CGIA        901     A
CGIA            SA6     IH2         IH2 = [II]                                          CGIA        902     A
CGIA            BX3     X5                                                              CGIA        903     A
CGIA            RJ      GAI         GET ADDR INFO                                       CGIA        904     A
CGIA            SB3     X0                                                              CGIA        905     A
CGIA            SB4     X3                                                              CGIA        906     A
CGIA            MX7     -18                                                             CGIA        907     A
CGIA            SX3     B3-B4       K = K - RA[RBWORD]                                  CGIA        908     A
CGIA            BX3     -X7*X3                                                          CGIA        909     A
CGIA            MX7     -12                                                             CGIA        910     A
CGIA            LX7     18                                                              CGIA        911     A
CGIA            BX4     -X7*X0                                                          CGIA        912     A
CGIA            LX0     30                                                              CGIA        913     A
CGIA            BX4     X3+X4                                                           CGIA        914     A
CGIA            SX6     X0                                                              CGIA        915     A
CGIA            MX7     0                                                               CGIA        916     A
CGIA            SA6     IH          RESET IH                                            CGIA        917     A
CGIA            SA7     RBWORD      RBWORD = 0    */ NO RELOCATION FOR THIS INS         CGIA        918     A
CGIA                                                                                    CGIA        919     A
CGIA      PLD4  WRTEXT 2            OUTPUT BINARY                                       CGIA        920     A
CGIA      .FPAS IFNE    HC.FPAS,0                                                       CGIA        921     A
CGIA      .FPAS ENDIF                                                                   CGIA        922     A
CGIA            EQ      GNIW                                                            CGIA        923     A
CGIA                                                                                    CGIA        924     A
CGIA      .TS   ENDIF                                                                   CGIA        925     A
CGIA      JUMPS SPACE   2,14                                                            CGIA        926     A
CGIA     **     JUMP PROCESSORS                                                         CGIA        927     A
CGIA      RJXJ  SPACE   2,14                                                            CGIA        928     A
CGIA      .RJXJ IFNE    HC.RJXJ,0                                                       CGIA        929     A
CGIA                                                                                    CGIA        930     A
CGIA     **     RJXJ   CA,IH,0,RI  ,  CA = 12/BIAS,6/JT                                CGIA        931     A
CGIA     *      OUTPUT BINARY FOR ^JT  XI,*+1 ;  RJ IH+CA                               CGIA        932     A
CGIA                                                                                    CGIA        933     A
CGIA            PROCESS RJXJ                                                            CGIA        934     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
      CGIA          RJ     FPU         FORCE UPPER                                  CGIA        935     A
      CGIA          SX1    B1                                                       CGIA        936     A
      CGIA          LX1    AR.RLP                                                   CGIA        937     A
      CGIA          SB2    2                                                        CGIA        938     A
      CGIA          RJ     ATR         SET UPPER TEXT RELOC FOR X-JUMP TO *+1       CGIA        939     A
      CGIA          MX0    -3                                                       CGIA        940     A
      CGIA          BX1    -X0*X5      I = RI[II] & 7                               CGIA        941     A
      CGIA          SX4    0300B                                                    CGIA        942     A
      CGIA          BX4    X1+X4       030I                                         CGIA        943     A
      CGIA          LX5    -SI.CAP                                                  CGIA        944     A
      CGIA          BX6    -X0*X5      JT = CA[II] & 7                              CGIA        945     A
      CGIA          SX7    B1                                                       CGIA        946     A
      CGIA          BX3    X6-X7       J = XOR(JT,1)  */ TOGGLE JUMP TYPE           CGIA        947     A
      CGIA          LX3    3                                                        CGIA        948     A
      CGIA          BX4    X3+X4                                                    CGIA        949     A
      CGIA          LX4    18                                                       CGIA        950     A
      CGIA          SA2    AI                                                       CGIA        951     A
      CGIA          SX3    X2+B1       K = ORGC[AI] + 1                             CGIA        952     A
      CGIA          BX4    X3+X4       03JIK                                        CGIA        953     A
      CGIA          LX4    6                                                        CGIA        954     A
      CGIA          BX4    X7+X4       ADD 01 OPCODE                                CGIA        955     A
      CGIA          LX4    24                                                       CGIA        956     A
      CGIA          SX3    X5                                                       CGIA        957     A
      CGIA          AX3    6                                                        CGIA        958     A
      CGIA          IX4    X3+X4       ADD BIAS                                     CGIA        959     A
      CGIA          LX5    SI.CAP-SI.IHP                                            CGIA        960     A
      CGIA          SA2    BCDC                                                     CGIA        961     A
      CGIA          ZR     X2,RJXJ2    IF BCDC = 0                                  CGIA        962     A
      CGIA                                                                          CGIA        963     A
      CGIA          SA3    =10H                                                     CGIA        964     A
      CGIA          SA2    =10HRJ::   X0,                                           CGIA        965     A
      CGIA          BX7    X3                                                       CGIA        966     A
      CGIA          LX1    6                                                        CGIA        967     A
      CGIA          SA7    LBUF                                                     CGIA        968     A
      CGIA          IX7    X1+X2       ADD REG NUMBER                               CGIA        969     A
      CGIA          SA3    FMI.XJ+X6                                                CGIA        970     A
      CGIA          MX6    12                                                       CGIA        971     A
      CGIA          BX6    X6*X3                                                    CGIA        972     A
      CGIA          LX6    -12                                                      CGIA        973     A
      CGIA          BX7    X6+X7                                                    CGIA        974     A
      CGIA          SA7    A7+B1                                                    CGIA        975     A
      CGIA          BX1    X4          SAVE BINARY                                  CGIA        976     A
      CGIA          SX2    X5                                                       CGIA        977     A
      CGIA          RJ     CSN         CONVERT NAME                                 CGIA        978         I
               -CCG16
      CGIA          CALL   CSN#        CONVERT NAME                                 CCG16        38     A
      CGIA          SB4    60                                                       CGIA        979     A
      CGIA          BX7    X6                                                       CGIA        980     A
      CGIA          SB4    B4-B3       UBC = 60 - BC                                CGIA        981     A
      CGIA          SX6    X1                                                       CGIA        982     A
      CGIA          ZR     X6,RJXJ1    IF BIAS = 0                                  CGIA        983     A
      CGIA          RJ     CON                                                      CGIA        984     A

               0         1         2         3         4         5         6         7         8
               12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA              SX3    1R+                                          CGIA        985     A
CGIA              LX3    -6                                           CGIA        986     A
CGIA              BX6    X3+X6                                        CGIA        987     A
CGIA              RJ     ADC         ADD +BIAS                        CGIA        988     A
CGIA      RJXJ1   RJ     TSB         TERMINATE STRING                 CGIA        989     A
CGIA              BX4    X1                                           CGIA        990     A
CGIA      RJXJ2   SX3    X5                                           CGIA        991     A
CGIA              RJ     GAI         GET ADDR INFO OF EXTERNAL        CGIA        992     A
CGIA              IX4    X3+X4                                        CGIA        993     A
CGIA              WRTEXT 4,RELOC     WRITE BINARY AND ADD RELOC       CGIA        994     A
CGIA              EQ     GNIW                                         CGIA        995     A
CGIA      .RJXJ   ENDIF                                               CGIA        996     A
CGIA      JPX     SPACE  2,14                                         CGIA        997     A
CGIA              PROCESS JPX,0300B  03IJK                            CGIA        998     A
CGIA              MX0    -3                                           CGIA        999     A
CGIA              BX2    -X0*X5      J = RI[II] & 7                   CGIA       1000     A
CGIA              LX5    -SI.CAP                                      CGIA       1001     A
CGIA              BX1    -X0*X5      I = CA[II] & 7  */ JUMP TYPE     CGIA       1002     A
CGIA                                                                  CGIA       1003     A
CGIA      JPX1    LX1    3                                            CGIA       1004     A
CGIA              BX2    X1+X2                                        CGIA       1005     A
CGIA              LX5    SI.CAP-SI.IHP                                CGIA       1006     A
CGIA              BX4    X4+X2                                        CGIA       1007     A
CGIA              LX4    18                                           CGIA       1008     A
CGIA              SX3    X5                                           CGIA       1009     A
CGIA              RJ     GAI                                          CGIA       1010     A
CGIA              IX4    X3+X4                                        CGIA       1011     A
CGIA              WRTEXT 2                                            CGIA       1012     A
CGIA              EQ     GNIW                                         CGIA       1013     A
CGIA                                                                  CGIA       1014     A
CGIA              PROCESS JPBB                                        CGIA       1015     A
CGIA              MX0    -3                                           CGIA       1016     A
CGIA              BX1    -X0*X5      I = RI[II] & 7                   CGIA       1017     A
CGIA              LX5    -SI.RJP                                      CGIA       1018     A
CGIA              BX2    -X0*X5      J = RJ[II] & 7                   CGIA       1019     A
CGIA              LX5    SI.RJP-SI.CAP                                CGIA       1020     A
CGIA              SX4    X5+4        FM = 04+CA[II]                   CGIA       1021     A
CGIA              LX4    6                                            CGIA       1022     A
CGIA              EQ     JPX1                                         CGIA       1023     A
CGIA      JIN     SPACE  2,14                                         CGIA       1024     A
CGIA              PROCESS JIN,0200B                                   CGIA       1025     A
CGIA              BX1    -X0*X5                                       CGIA       1026     A
CGIA              LX1    3                                            CGIA       1027     A
CGIA              BX4    X1+X4       02I0                             CGIA       1028     A
CGIA              LX5    -SI.IHP                                      CGIA       1029     A
CGIA              SX5    X5                                           CGIA       1030     A
CGIA                                                                  CGIA       1031     A
CGIA              PROCESS RJ3,0100B                                   CGIA       1032     A
CGIA              PROCESS UJP,0400B                                   CGIA       1033     A
CGIA              SX3    X5                                           CGIA       1034     A
CGIA              LX4    18                                           CGIA       1035     A
CGIA              RJ     GAI                                          CGIA       1036     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                LX5    -SI.CAP                                          CGIA        1037     A
CGIA                SX5    X5                                               CGIA        1038          I
          -CCGA072
CGIA                IX3    X3+X5                                            CGIA        1039          I
          -CCGA072
CGIA                SX7    X5                                               CCGA072       16     A
CGIA                SA7    IH2+1       SAVE FOR LISTING                     CCGA072       17     A
CGIA                IX3    X3+X7                                            CCGA072       18     A
CGIA                MX0    -18                                              CGIA        1040     A
CGIA                BX3    -X0*X3                                           CGIA        1041     A
CGIA                IX4    X3+X4                                            CGIA        1042     A
CGIA                WRTEXT 2                                                CGIA        1043     A
CGIA                RJ     FPU         FORCE UPPER AFTER                    CGIA        1044     A
CGIA                EQ     MPXA                                             CGIA        1045     A
CGIA                                                                        CGIA        1046     A
CGIA      RJ6       SPACE  2,14                                             CGIA        1047     A
CGIA      .HC       IFNE   HC.RJ6,0                                         CGIA        1048     A
CGIA                                                                        CGIA        1049     A
CGIA      RJTA      PSEUDO RJT,(S,C)                                        CGIA        1050     A
CGIA      RJTB      SB0    0                                                CGIA        1051     A
CGIA                SB0    0                                                CGIA        1052     A
CGIA                                                                        CGIA        1053     A
CGIA                PROCESS RJ6                                             CGIA        1054     A
CGIA                SA1    RJTA                                             CGIA        1055     A
CGIA                UX5    X5                                               CGIA        1056     A
CGIA                LX5    30                                               CGIA        1057     A
CGIA                RJ     LPO         LIST PSEUDO                          CGIA        1058     A
CGIA                LX5    30                                               CGIA        1059     A
CGIA                RJ     FPU         FORCE UPPER                          CGIA        1060     A
CGIA                SX3    X5                                               CGIA        1061     A
CGIA                RJ     GAI         GET ADDR INFO                        CGIA        1062     A
CGIA      .FPAS     IFNE   HC.FPAS,0                                        CGIA        1063     A
CGIA      RJT1      BSS    0                                                CGIA        1064     A
CGIA      .FPAS     ENDIF                                                   CGIA        1065     A
CGIA                SB2    2                                                CGIA        1066     A
CGIA                RJ     ATR         ADD UPPER RELOC FOR THE NAME         CGIA        1067     A
CGIA                SX4    B1                                               CGIA        1068     A
CGIA                LX4    54                                               CGIA        1069     A
CGIA      .RJTBN    IFC    NE,/"HC.RJTBN"//                                 CGIA        1070     A
CGIA                SA3    =X"HC.RJTBN"                                     CGIA        1071     A
CGIA                RJ     GAI                                              CGIA        1072     A
CGIA                BX4    X3+X4       ADD RA OF TRACEBACK NAME             CGIA        1073     A
CGIA      .RJTBN    ENDIF                                                   CGIA        1074     A
CGIA                LX5    -SI.CAP                                          CGIA        1075     A
CGIA                SX6    X5-7777B                                         CGIA        1076     A
CGIA                PL     X6,RJT2     IF CA[II] \ 7777B                    CGIA        1077     A
CGIA                SX7    X5                                               CGIA        1078     A
CGIA                LX7    18                                               CGIA        1079     A
CGIA                BX4    X4+X7       12/0100,18/0,12/CA,18/TEMPA0.        CGIA        1080     A
CGIA                WRTEXT 4,"HC.RJTBN"      WRITE BINARY AND ADD RELOC      CGIA        1081     A
CGIA                EQ     MPXA                                             CGIA        1082     A
CGIA                                                                        CGIA        1083     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA      RJT2     MX7    -12                                                    CGIA        1084     A
CGIA               LX7    18                                                     CGIA        1085     A
CGIA               BX4    -X7+X4                                                 CGIA        1086     A
CGIA               WRTEXT 4,"HC.RJTBN"                                           CGIA        1087     A
CGIA               SA5    A5                                                     CGIA        1088     A
CGIA               SA4    RJTB                                                   CGIA        1089     A
CGIA               LX5    -SI.CAP                                                CGIA        1090     A
CGIA               SX6    X5                                                     CGIA        1091     A
CGIA               BX4    X4+X6       OUTPUT SB0  0  ;  SB0 LINE-NUM             CGIA        1092     A
CGIA               WRTEXT 4                                                      CGIA        1093     A
CGIA               EQ     MPXA                                                   CGIA        1094     A
CGIA                                                                             CGIA        1095     A
CGIA      .HC      ENDIF                                                          CGIA        1096     A
CGIA      BMI      SPACE  3,14                                                   CGIA        1097     A
CGIA      .TS      IFNE   HC.TS,0                                                CGIA        1098     A
CGIA                                                                             CGIA        1099     A
CGIA     **        BMI - BEGIN *TS* FORMAT MACHINE INSTRUCTION PROCESSING        CGIA        1100     A
CGIA                                                                             CGIA        1101     A
CGIA               PROCESS BMI                                                   CGIA        1102     A
CGIA               SX6    BMI1                                                   CGIA        1103     A
CGIA               EQ     SXA         SET EXIT ADDR AND START PROCESSING         CGIA        1104     A
CGIA                                                                             CGIA        1105     A
CGIA      BMI1     GNIW               WD = GNIW( INPUT )                         CGIA        1106     A
CGIA               SX7    B0                                                     CGIA        1107     A
CGIA               SA7    IH          IH = 0                                     CGIA        1108     A
CGIA               MX0    6                                                      CGIA        1109     A
CGIA               BX3    X0*X5                                                  CGIA        1110     A
CGIA               LX3    6           OPC = SHIFT( MASK(6) & WD , 6 )            CGIA        1111     A
CGIA               ZR     X3,BMI6     IF OPC = 0    */ PSEUDO OP                 CGIA        1112     A
CGIA               SX7    X3-10B                                                 CGIA        1113     A
CGIA               SX4    X3-50B                                                 CGIA        1114     A
CGIA               MI     X7,BMI2     IF OPC < 10B  */ 30 BIT INSTRUCTIONS       CGIA        1115     A
CGIA               MI     X4,BMI3     IF OPC < 50B  */ SHORT INSTRUCTIONS        CGIA        1116     A
CGIA               MX0    -3                                                     CGIA        1117     A
CGIA               BX2    -X0*X3                                                 CGIA        1118     A
CGIA               SX4    X2-3                                                   CGIA        1119     A
CGIA               PL     X4,BMI3     IF OPC & 7 > 2  */ SHORT INSTRUCTION       CGIA        1120     A
CGIA                                                                             CGIA        1121     A
CGIA     *         PROCESS 30 BIT INSTRUCTION - 12/FMIJ,18/CA,12/0,18/IH         CGIA        1122     A
CGIA                                                                             CGIA        1123     A
CGIA      BMI2     SX3    X5                                                     CGIA        1124     A
CGIA               LX5    30                                                     CGIA        1125     A
CGIA               RJ     GAI         GET ADDR INFO                              CGIA        1126     A
CGIA               SX6    X5                                                     CGIA        1127     A
CGIA               SX6    IH2+1       IH2(2) = CA                                CGIA        1128     A
CGIA               IX7    X6+X3       K = CA + RA[RBWORD]                        CGIA        1129     A
CGIA               MX0    -18                                                    CGIA        1130     A
CGIA               BX7    -X0*X7                                                 CGIA        1131     A
CGIA               MX1    -12                                                    CGIA        1132     A
CGIA               LX1    18                                                     CGIA        1133     A
CGIA               BX3    -X1*X5      FMIJ 0                                     CGIA        1134     A
CGIA               BX4    X3+X7                                                  CGIA        1135     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA              WRTEXT 2          OUTPUT BINARY, ETC                          CGIA      1136    A
CGIA              EQ     BMI1                                                   CGIA      1137    A
CGIA                                                                            CGIA      1138    A
CGIA     *        PROCESS UP TO 4 15 BIT INSTRUCTIONS                            CGIA      1139    A
CGIA                                                                            CGIA      1140    A
CGIA      BMI3    BX0    X5                                                     CGIA      1141    A
CGIA                                                                            CGIA      1142    A
CGIA      BMI4    LX0    15         WD = SHIFT(WD,15)                           CGIA      1143    A
CGIA              MX0    -15                                                    CGIA      1144    A
CGIA              BX4    -X1*X0     DATA = WD & 77777B                          CGIA      1145    A
CGIA              BX0    X1*X0      WD = WD & MASK(45)  */ REMOVE BYTE          CGIA      1146    A
CGIA      BMI5    WRTEXT 1                                                      CGIA      1147    A
CGIA                                                                            CGIA      1148    A
CGIA              NZ     X0,BMI4    IF WD " 0                                   CGIA      1149    A
CGIA              EQ     BMI1                                                   CGIA      1150    A
CGIA                                                                            CGIA      1151    A
CGIA     *        PROCESS PSEUDO INSTRUCTION - 6/0,60N,18/CA,12/H2,18/IH        CGIA      1152    A
CGIA                                                                            CGIA      1153    A
CGIA      BMI6    BX3    X5                                                     CGIA      1154    A
CGIA              AX3    -12                                                    CGIA      1155    A
CGIA              SB3    X3+                                                    CGIA      1156    A
CGIA              JP     BMI.JT+B3                                              CGIA      1157    A
CGIA                                                                            CGIA      1158    A
CGIA      BMI.JT  BSS    0                                                      CGIA      1159    A
CGIA              LOC    0                                                      CGIA      1160    A
CGIA              SX6    GNIW       0 - EMI - END MACHINE INST PROCESSING       CGIA      1161    A
CGIA              EQ     SXA                                                    CGIA      1162    A
CGIA              EQ     .BOS                                                   CGIA      1163    A
CGIA              SX4    0400B                                                  CGIA      1164    A
CGIA              EQ     .UJP                                                   CGIA      1165    A
CGIA              SX4    0100B                                                  CGIA      1166    A
CGIA              EQ     .RJ3                                                   CGIA      1167    A
CGIA              EQ     .RJ6                                                   CGIA      1168    A
CGIA              UX4    X5         LCM READ/WRITE - 12/00NN,33/0,15/01XJK      CGIA      1169    A
CGIA              MX0    0                                                      CGIA      1170    A
CGIA              EQ     BMI5                                                   CGIA      1171    A
CGIA     *               ** NEED JIN, BSS, OTR PROCESSORS                      CGIA      1172    A
CGIA              LOC    *O                                                     CGIA      1173    A
CGIA      SXA     SPACE  2,14                                                   CGIA      1174    A
CGIA     **       SXA - SET EXIT ADDRESS OF PROCESSORS USED BY BOTH *SI.*       CGIA      1175    A
CGIA     *               AND *TS* FORMAT PROCESSING                            CGIA      1176    A
CGIA     *                                                                     CGIA      1177    A
CGIA     *        ENTRY  (X6) = EXIT ADDRESS                                   CGIA      1178    A
CGIA     *                                                                     CGIA      1179    A
CGIA     *        EXITS TO (X6)                                                CGIA      1180    A
CGIA                                                                            CGIA      1181    A
CGIA      SXA     SB3    X6                                                     CGIA      1182    A
CGIA              LX6    32                                                     CGIA      1183    A
CGIA              PX7    X6                                                     CGIA      1184    A
CGIA              AX7    2                                                      CGIA      1185    A
CGIA              SA7    MPXA       MPXA = 0400ADDR,0                           CGIA      1186    A
CGIA              JP     B3                                                     CGIA      1187    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                                                                    CGIA        1188     A
CGIA      MPXA     EQ     GNIW        MIXED PROCESSOR EXIT ADDRESS WORD  CGIA        1189     A
CGIA                                                                    CGIA        1190     A
CGIA      .TS      ELSE                                                 CGIA        1191     A
CGIA                                                                    CGIA        1192     A
CGIA      MPXA     =      GNIW        NO TS MACHINE INSTRUCTION PROCESSING  CGIA     1193     A
CGIA                                                                    CGIA        1194     A
CGIA      .TS      ENDIF                                                CGIA        1195     A
CGIA      LCC      TITLE  INITIAL CARD GROUP PROCESSING AND INITIALIZATION  CGIA     1196     A
CGIA     **        LCC    0,WC AND WC WORDS FOLLOW IN *C* FORMAT        CGIA        1197     A
CGIA                                                                    CGIA        1198     A
CGIA               PROCESS LCC                                          CGIA        1199     A
CGIA               ALLOC  LCC,X5+B1   ALLOC( LCC , WC+1 )              CGIA        1200     A
CGIA               BX6    X5                                           CGIA        1201     A
CGIA               SA6    X2+B6                                        CGIA        1202     A
CGIA               SA0    A6+1                                         CGIA        1203     A
CGIA               RJ     MWI         MWI( WC , LCC )                  CGIA        1204     A
CGIA               EQ     GNIW                                         CGIA        1205     A
CGIA      IDENT    SPACE  2,10                                         CGIA        1206     A
CGIA     **        IDENT  0,SY                                         CGIA        1207     A
CGIA                                                                    CGIA        1208     A
CGIA               PROCESS IDENT                                        CGIA        1209     A
CGIA               SX3    X5                                           CGIA        1210     A
CGIA               RJ     GAI                                          CGIA        1211     A
CGIA               SA1    A1-2        WORD A                           CGIA        1212     A
CGIA               MX0    42                                           CGIA        1213     A
CGIA               BX6    X0*X1                                        CGIA        1214     A
CGIA               SA6    ERRL+2      SAVE NAME IN *ERR* LINE          CGIA        1215     A
CGIA               SA6    NAE+3       SAVE IN ASSEMBLY ERRORS MESSAGE  CGIA        1216     A
CGIA               SA6    IDT+1       SAVE NAME IN 77 TABLE            CGIA        1217     A
CGIA               MOVE   2,=XHO$DATE,A6+B1 DATE AND TIME              CGIA        1218          I
          -CCGA002
CGIA               SA1    =XHO$DATE                                    CCGA002        1     A
CGIA               SA2    A1+B1                                        CCGA002        2     A
CGIA               LX1    6                                            CCGA002        3     A
CGIA               LX2    6                                            CCGA002        4     A
CGIA               BX6    X1                                           CCGA002        5     A
CGIA               BX7    X2                                           CCGA002        6     A
CGIA               SA6    A6+B1                                        CCGA002        7     A
CGIA               SA7    A6+B1                                        CCGA002        8     A
CGIA               MOVE   3,=XHO$CCOP,IDT+CCO  CONTROL CARD OPTIONS    CGIA        1219     A
CGIA               EQ     GNIW                                         CGIA        1220     A
CGIA      TITLE    SPACE  2,12                                         CGIA        1221     A
CGIA     **        TITLE  0,WC AND WC WORDS FOLLOW                     CGIA        1222     A
CGIA                                                                    CGIA        1223     A
CGIA               PROCESS TITLE                                        CGIA        1224     A
CGIA               SA4    =1H                                          CGIA        1225     A
CGIA               SA0    =XF$STITL+1                                  CGIA        1226     A
CGIA               BX6    X4                                           CGIA        1227     A
CGIA               SA6    A0-B1       O.STITL(1) = 10H                 CGIA        1228     A
CGIA               SX6    X5+2                                         CGIA        1229     A
CGIA               SA6    =XN$STITL   N$STITL = WC + 2                 CGIA        1230     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA              RJ     MWI         MWI( WC , O.STITL+1 )                          CGIA        1231     A
CGIA              MX6    0                                                          CGIA        1232     A
CGIA              SA6    A0          LINE TERMINATOR                                CGIA        1233     A
CGIA              EQ     GNIW                                                       CGIA        1234     A
CGIA      COMMENT SPACE   2,12                                                      CGIA        1235     A
CGIA     *        COMNT  0,WC AND WC WORDS FOLLOW IN *H* FORMAT                     CGIA        1236     A
CGIA                                                                                CGIA        1237     A
CGIA              PROCESS COMNT                                                     CGIA        1238     A
CGIA              SA0    IDT+COMMNT                                                 CGIA        1239     A
CGIA              RJ     MWI         MWI( WC , IDT+COMMENT )                        CGIA        1240     A
CGIA              EQ     GNIW                                                       CGIA        1241     A
CGIA      LIB     SPACE  2                                                          CGIA        1242     A
CGIA     *        LIB - DEFINE LIBRARY NAME                                         CGIA        1243     A
CGIA                                                                                CGIA        1244     A
CGIA              PROCESS LIB                                                       CGIA        1245     A
CGIA              UX1    X5                                                         CGIA        1246     A
CGIA              LX1    12                                                         CGIA        1247     A
CGIA              ADDWRD LIB,X1      SAVE NAME IN *LIB*                             CGIA        1248     A
CGIA              EQ     GNIW                                                       CGIA        1249     A
CGIA      MWI     SPACE  3,14                                                       CGIA        1250     A
CGIA     **       MWI - MOVE WORDS FROM INPUT BUFFER TO WORKING STORAGE             CGIA        1251     A
CGIA     *                                                                          CGIA        1252     A
CGIA     *        ENTRY  (X5) = WC                                                  CGIA        1253     A
CGIA     *               (A0) = FWA OF WS BUFFER                                    CGIA        1254     A
CGIA     *                                                                          CGIA        1255     A
CGIA     *        EXIT   (A0) = LWA+1                                               CGIA        1256     A
CGIA     *               (A7,X7) = LWA AND [LWA]                                    CGIA        1257     A
CGIA                                                                                CGIA        1258     A
CGIA      MWI1    GNIW               GET WORD FROM INPUT                            CGIA        1259     A
CGIA              BX7    X5                                                         CGIA        1260     A
CGIA              SA7    A0          STORE IN WS BUFFER                             CGIA        1261     A
CGIA              SA0    A0+B1       FWA = FWA + 1                                  CGIA        1262     A
CGIA              SX0    X0-1        WC = WC - 1                                    CGIA        1263     A
CGIA              NZ     X0,MWI1     IF WC " 0                                      CGIA        1264     A
CGIA      MWI     ROUTINE                                                           CGIA        1265     A
CGIA              SX0    X5                                                         CGIA        1266     A
CGIA              EQ     MWI1                                                       CGIA        1267     A
CGIA      USEBLK  SPACE  3,14                                                       CGIA        1268     A
CGIA     **       USBLK - TERMINATE INITIAL CARD GROUP PROCESSING                   CGIA        1269     A
CGIA                                                                                CGIA        1270     A
CGIA              PROCESS USBLK                                                     CGIA        1271     A
CGIA              SX6    A5                                                         CGIA        1272     A
CGIA              SA6    IBUF        IBUF = II     */ SAVE INPUT BUFFER POINTER     CGIA        1273     A
CGIA              ADDWRD LIB,0       TERMINATE LIB TABLE                            CGIA        1274     A
CGIA              ADDWRD LCC,0       TERMINATE LCC TABLE                            CGIA        1275     A
CGIA              SA1    HO$B                                                       CGIA        1276     A
CGIA              ZR     X1,USBLK4   IF B = 0                                       CGIA        1277     A
CGIA                                                                                CGIA        1278     A
CGIA     *        OUTPUT *LCC* DIRECTIVES TO *LGO* AS SEPERATE RECORDS              CGIA        1279     A
CGIA                                                                                CGIA        1280     A
CGIA              BX5    X2          (X5) = O.LCC                                   CGIA        1281     A
CGIA      USBLK1  SA4    X5          WORD COUNT                                     CGIA        1282     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                ZR     X4,USBLK2   IF [FWA] = 0  */ END OF LCC TABLE      CGIA    1283    A
CGIA                SB6    X5+B1                                              CGIA    1284    A
CGIA                SX5    B6+X4       FWA = FWA + 1+[FWA]                    CGIA    1285    A
CGIA                WRLGO  B6,X4       LCC DIRECTIVE TO LGO                   CGIA    1286    A
CGIA                WRITER F.LGO,,R                                           CGIA    1287    A
CGIA                EQ     USBLK1                                             CGIA    1288    A
CGIA                                                                          CGIA    1289    A
CGIA       USBLK2   WRLGO  IDT,15      OUTPUT 77 TABLE                        CGIA    1290    A
CGIA                                                                          CGIA    1291    A
CGIA       USBLK4   CALL   CG$ASN      ADJUST SYMTAB                          CGIA    1292    A
CGIA                                                                          CGIA    1293    A
CGIA     *          SCAN *SYM* AND APPEND A $ TO NAMES THAT CONFLICT WITH     CGIA    1294    A
CGIA     *          REGISTER NAMES                                            CGIA    1295    A
CGIA                                                                          CGIA    1296    A
CGIA                SA1    O.SYM                                              CGIA    1297    A
CGIA                SA2    L.SYM                                              CGIA    1298    A
CGIA                SA1    X1                                                 CGIA    1299    A
CGIA                MX0    -6                                                 CGIA    1300    A
CGIA                SB4    1RX                                                CGIA    1301    A
CGIA                SB6    6                                                  CGIA    1302    A
CGIA                SX5    1R$                                                CGIA    1303    A
CGIA                SB3    B1+B1                                              CGIA    1304    A
CGIA                LX5    42                                                 CGIA    1305    A
CGIA                SX4    B6                                                 CGIA    1306    A
CGIA                BX5    X4+X5                                              CGIA    1307    A
CGIA                                                                          CGIA    1308    A
CGIA       USBLK4A  SX2    X2-3                                               CGIA    1309    A
CGIA                ZR     X2,USBLK4C  IF FINISHED                            CGIA    1310    A
CGIA                SA1    A1+3        NEXT NAME                              CGIA    1311    A
CGIA                BX3    -X0*X1                                             CGIA    1312    A
CGIA                LX6    B6,X1                                              CGIA    1313    A
CGIA                SX4    X3-12                                              CGIA    1314    A
CGIA                NZ     X4,USBLK4A  IF NOT A 2 CHAR NAME                   CGIA    1315    A
CGIA                BX7    -X0*X6                                             CGIA    1316    A
CGIA                SB2    X7                                                 CGIA    1317    A
CGIA                LE     B2,B3,USBLK4B     IF FIRST CHAR IS A OR B          CGIA    1318    A
CGIA                NE     B2,B4,USBLK4A     IF NOT X                         CGIA    1319    A
CGIA       USBLK4B  LX6    6                                                  CGIA    1320    A
CGIA                BX7    -X0*X6                                             CGIA    1321    A
CGIA                SX6    X7-1R0                                             CGIA    1322    A
CGIA                MI     X6,USBLK4A                                         CGIA    1323    A
CGIA                SX6    X7-1R8                                             CGIA    1324    A
CGIA                PL     X6,USBLK4A                                         CGIA    1325    A
CGIA                                                                          CGIA    1326    A
CGIA                IX6    X5+X1       APPEND $ TO THE NAME                   CGIA    1327    A
CGIA                SA6    A1                                                 CGIA    1328    A
CGIA                EQ     USBLK4A                                            CGIA    1329    A
CGIA                                                                          CGIA    1330    A
CGIA       USBLK4C  SA5    BCDC                                               CGIA    1331    A
CGIA                ZR     X5,USBLK9A  IF BCDC = 0   */ OL & C = 0            CGIA    1332    A
CGIA                                                                          CGIA    1333    A
CGIA                NUPAGE             FORCE EJECT AND OUTPUT TITLE           CGIA    1334    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                SA2    =8LIDENT                                          CGIA        1335     A
CGIA                SA3    IDT+1                                             CGIA        1336     A
CGIA                RJ     FSP         OUTPUT IDENT LINE                     CGIA        1337     A
CGIA                SA5    BCDC                                              CGIA        1338     A
CGIA                PL     X5,USBLK5   IF C = 0                              CGIA        1339     A
CGIA                                                                         CGIA        1340     A
CGIA                SA3    N$STITL                                           CGIA        1341     A
CGIA                SA4    =10H  TITLE                                       CGIA        1342     A
CGIA                BX6    X4                                                CGIA        1343     A
CGIA                SA6    F$STITL                                           CGIA        1344     A
CGIA                SB7    X3                                                CGIA        1345     A
CGIA                LE     B7,B1,USBLK5      IF L.STITL @ 1                  CGIA        1346     A
CGIA                WRITEC F.CMPS,A6,B7                                      CGIA        1347     A
CGIA                                                                         CGIA        1348     A
CGIA     *          LIST *LCC* DIRECTIVES                                    CGIA        1349     A
CGIA                                                                         CGIA        1350     A
CGIA      USBLK5    SA4    O.LCC                                             CGIA        1351     A
CGIA                SA5    X4                                                CGIA        1352     A
CGIA                SB5    X5          WC = [LCC]                            CGIA        1353     A
CGIA                ZR     X5,USBLK7   IF WC = 0                             CGIA        1354     A
CGIA                SA2    =1H                                               CGIA        1355     A
CGIA                SA3    =8LLCC                                            CGIA        1356     A
CGIA                SB4    12          UBC = 12                              CGIA        1357     A
CGIA                BX6    X2                                                CGIA        1358     A
CGIA                SA6    LBUF                                              CGIA        1359     A
CGIA                BX7    X3                                                CGIA        1360     A
CGIA      USBLK6    SA5    A5+B1                                             CGIA        1361     A
CGIA                SB3    60                                                CGIA        1362     A
CGIA                BX6    X5                                                CGIA        1363     A
CGIA                RJ     ADC                                               CGIA        1364     A
CGIA                NZ     B5,USBLK6                                         CGIA        1365     A
CGIA                SX6    A5+B1                                             CGIA        1366     A
CGIA                SA6    O.LCC                                             CGIA        1367     A
CGIA                RJ     TSB         TERMINATE THE LINE                    CGIA        1368     A
CGIA                WRTEXT 0           AND LIST IT                           CGIA        1369     A
CGIA                EQ     USBLK5                                            CGIA        1370     A
CGIA                                                                         CGIA        1371     A
CGIA      USBLK7    SA1    O.LCC                                             CGIA        1372     A
CGIA                SA2    L.LCC                                             CGIA        1373     A
CGIA                SX2    X2-1                                              CGIA        1374     A
CGIA                IX6    X1-X2       RESET O.LCC                           CGIA        1375     A
CGIA                SA6    A1                                                CGIA        1376     A
CGIA                                                                         CGIA        1377     A
CGIA     *          LIST THE LIB DIRECTIVES                                  CGIA        1378     A
CGIA                                                                         CGIA        1379     A
CGIA                SA5    O.LIB                                             CGIA        1380     A
CGIA                SX0    X5+2                                              CGIA        1381     A
CGIA      USBLK8    SA2    =8LLIB=                                           CGIA        1382     A
CGIA                SA3    X0                                                CGIA        1383     A
CGIA                SX0    X0+B1                                             CGIA        1384     A
CGIA                ZR     X3,USBLK9   IF END OF THE TABLE                   CGIA        1385     A
CGIA                RJ     FSP         FORMAT AND LIST  LIB=  LIBNAME        CGIA        1386     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                EQ     USBLK8                                                  CGIA        1387     A
CGIA                                                                               CGIA        1388     A
CGIA       USBLK9   SA2    =8LUSEBLK                                               CGIA        1389     A
CGIA                SA3    =1H                                                     CGIA        1390     A
CGIA                RJ     FSP          FORMAT AND LIST *USEBLK* LINE              CGIA        1391     A
CGIA       USBLK9A  SA1    O.LIB                                                   CGIA        1392     A
CGIA                SA2    L.LIB                                                   CGIA        1393     A
CGIA                SB6    X1                                                      CGIA        1394     A
CGIA                SB7    X2-1                                                    CGIA        1395     A
CGIA                SB2    B1+B1                                                   CGIA        1396     A
CGIA                EQ     B7,B2,USBLK10     IF L.LIB = 3                          CGIA        1397     A
CGIA                SX6    LT.LDSET                                                CGIA        1398     A
CGIA                LX6    12                                                      CGIA        1399     A
CGIA                SX4    B7-B1                                                   CGIA        1400     A
CGIA                BX6    X4+X6                                                   CGIA        1401     A
CGIA                LX6    36                                                      CGIA        1402     A
CGIA                SA6    B6           HEADER WORD                                CGIA        1403     A
CGIA                SX7    LS.LIB                                                  CGIA        1404     A
CGIA                LX7    12                                                      CGIA        1405     A
CGIA                SX4    X4-1                                                    CGIA        1406     A
CGIA                BX7    X7+X4                                                   CGIA        1407     A
CGIA                LX7    36                                                      CGIA        1408     A
CGIA                SA7    A6+B1        LIB SUBTABLE HEADER WORD                   CGIA        1409     A
CGIA                WRLGO  B6,B7        OUTPUT LIB TABLE TO *LGO*                  CGIA        1410     A
CGIA                                                                               CGIA        1411     A
CGIA      *         SCAN *CBT* AND SETUP THE *PIDL* TABLE                          CGIA        1412     A
CGIA                                                                               CGIA        1413     A
CGIA       USBLK10  SA5    L.CBT                                                   CGIA        1414     A
CGIA                AX5    1                                                       CGIA        1415     A
CGIA                MX7    0            L.LCC = 0                                  CGIA        1416     A
CGIA                SA7    L.LCC                                                   CGIA        1417     A
CGIA                ALLOC  PIDL,X5+2    ALLOC( PIDL , L.CBT/2+2 )                  CGIA        1418     A
CGIA                SB6    X2                                                      CGIA        1419     A
CGIA                SB7    B1+B1        INDEX                                      CGIA        1420     A
CGIA                SA1    O.CBT                                                   CGIA        1421     A
CGIA                SA2    X1                                                      CGIA        1422     A
CGIA                SA3    X1+B1                                                   CGIA        1423     A
CGIA                MX0    -24                                                     CGIA        1424     A
CGIA                SB5    B1           PD = 1        */ PIDL TBL DECRMENT         CGIA        1425     A
CGIA                ZR     X3,USBLK14   IF LEN = 0    */ // COMM NOT REFED        CGIA        1426     A
CGIA                SB5    B1+B1        PD = 2                                     CGIA        1427     A
CGIA                SX6    B5+B1                                                   CGIA        1428     A
CGIA                SA6    CCB          CCB = 3                                    CGIA        1429     A
CGIA                EQ     USBLK12                                                 CGIA        1430     A
CGIA                                                                               CGIA        1431     A
CGIA       USBLK11  SA2    A2+2         FETCH NEXT ENTRY                           CGIA        1432     A
CGIA                SA3    A3+2                                                    CGIA        1433     A
CGIA       USBLK12  BX4    -X0*X3                                                  CGIA        1434     A
CGIA                PL     X3,USBLK13   IF ^LCM RESIDENT                           CGIA        1435     A
CGIA                MX7    -3                                                      CGIA        1436     A
CGIA                IX4    X4-X7                                                   CGIA        1437     A
CGIA                AX4    3            ROUND UP TO NEAREST 8                      CGIA        1438     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                SX7    B1                                                  CGIA        1439     A
CGIA                LX7    17                                                  CGIA        1440     A
CGIA                BX4    X7+X4       SET LCM BIT                             CGIA        1441     A
CGIA      USBLK13   BX6    X2+X4                                               CGIA        1442     A
CGIA                SA6    B6+B7       STORE ENTRY IN PIDL                     CGIA        1443     A
CGIA                SB7    B7+B1       BUMP INDEX                              CGIA        1444     A
CGIA      USBLK14   SX5    X5-1                                                CGIA        1445     A
CGIA                NZ     X5,USBLK11  IF NOT FINISHED                         CGIA        1446     A
CGIA                                                                           CGIA        1447     A
CGIA                SX7    B7-B5       L.PIDL = INDEX - PD                     CGIA        1448     A
CGIA                SX6    B6+B5                                               CGIA        1449     A
CGIA                SA7    L.PIDL                                              CGIA        1450     A
CGIA                SA6    O.PIDL                                              CGIA        1451     A
CGIA                SX7    B7-B1       LEN = INDEX - 1                         CGIA        1452     A
CGIA                LX7    36                                                  CGIA        1453     A
CGIA                SB2    LT.PIDL-2000B                                       CGIA        1454     A
CGIA                PX6    B2,X7                                               CGIA        1455     A
CGIA                SA6    B6          STORE HEADER                            CGIA        1456     A
CGIA                SA3    N$SLBT                                              CGIA        1457          I
          -CCG16
CGIA                SA3    =XN$SLBT                                            CCG16         39     A
CGIA                SA4    IDT+1                                               CGIA        1458     A
CGIA                BX6    X3+X4                                               CGIA        1459     A
CGIA                SA6    B6+B1       STORE PROG NAME AND LENGTH              CGIA        1460     A
CGIA                                                                           CGIA        1461     A
CGIA                WRLGO  B6,B7       PIDL TABLE TO LGO                       CGIA        1462     A
CGIA                                                                           CGIA        1463     A
CGIA      *         SETUP *LBIT*                                               CGIA        1464     A
CGIA                                                                           CGIA        1465     A
CGIA                SA5    L.LBT                                               CGIA        1466          I
          -CCG16
CGIA                SX5    =XZ$LBT                                             CCG16         40          I
          -CCGA072
CGIA                SA5    =XN$LBT                                             CCGA072       19     A
CGIA                LX1    B1,X5                                               CGIA        1467     A
CGIA                ALLOC  LBIT,X1     ALLOC( LBIT , L.LBT )                   CGIA        1468          I
          -CCG16
CGIA                SA4    O.LBT                                               CGIA        1469          I
          -CCG16
CGIA                ALLOC  LBIT,X1     ALLOC( LBIT, 2*Z$LBT )                  CCG16         41          I
          -CCGA072
CGIA                ALLOC  LBIT,X1     ALLOC( LBIT , 2*N$LBT )                 CCGA072       20     A
CGIA                SX0    B1                                                  CGIA        1470     A
CGIA                LX0    AD.RBP                                              CGIA        1471     A
CGIA                SB2    4                                                   CGIA        1472     A
CGIA                PX0    B2,X0                                               CGIA        1473     A
CGIA                SA4    X4                                                  CGIA        1474          I
          -CCG16
CGIA                SA4    =XF$LBT                                             CCG16         42     A
CGIA      USBLK16   SX4    X4          FWA                                     CGIA        1475     A
CGIA                BX6    X4+X0                                               CGIA        1476     A
CGIA                SA6    X2                                                  CGIA        1477     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                MX7    0                                                    CGIA        1478     A
CGIA                SA7    X2+B1                                                CGIA        1479     A
CGIA                SX5    X5-1                                                 CGIA        1480     A
CGIA                SX2    A7+B1                                                CGIA        1481     A
CGIA                SA4    A4+B1                                                CGIA        1482     A
CGIA                NZ     X5,USBLK16                                           CGIA        1483     A
CGIA                                                                            CGIA        1484     A
CGIA      *         LIST THE BLOCKS ( NAME TYPE ADDR  LENGTH )                   CGIA        1485     A
CGIA                                                                            CGIA        1486     A
CGIA                SA1    BCDC                                                 CGIA        1487     A
CGIA                SX1    X1+                                                  CGIA        1488     A
CGIA                ZR     X1,USBLK28  IF OL = 0 OR C " 0                       CGIA        1489     A
CGIA                MOVE   4,BLKA,LBUF                                          CGIA        1490     A
CGIA                LISTL  (=8L ),1    BLANK LINE                               CGIA        1491     A
CGIA                LISTL  OBUF,8      BLOCK MAP TITLE                          CGIA        1492     A
CGIA                SA1    =10HLOCAL                                            CGIA        1493     A
CGIA                BX6    X1                                                   CGIA        1494     A
CGIA                SA6    LBUF+1                                               CGIA        1495     A
CGIA                SA4    O.LBT                                                CGIA        1496          I
          -CCG16
CGIA                SA3    L.LBT                                                CGIA        1497          I
          -CCG16
CGIA                SX0    X3-1        NBLOCKS = L.LBT - 1                      CGIA        1498          I
          -CCG16
CGIA                SA5    X4                                                   CGIA        1499          I
          -CCG16
CGIA                SA5    F$LBT                                                CCG16         43     A
CGIA                SX0    Z$LBT       NBLOCKS = Z$LBT                          CCG16         44          I
          -CCGA072
CGIA                SA1    N$LBT                                                CCGA072       21     A
CGIA                SX0    X1                                                   CCGA072       22     A
CGIA                SA0    LBNT                                                 CGIA        1500     A
CGIA                                                                            CGIA        1501     A
CGIA      USBLK18   SA4    A0          10H NAME                                 CGIA        1502     A
CGIA                SA0    A0+B1                                                CGIA        1503     A
CGIA                SX1    X5          FWA                                      CGIA        1504     A
CGIA                BX6    X4                                                   CGIA        1505     A
CGIA                SA6    LBUF                                                 CGIA        1506     A
CGIA                CALL   COD=                                                 CGIA        1507     A
CGIA                LX6    24                                                   CGIA        1508     A
CGIA                SA6    LBUF+2                                               CGIA        1509     A
CGIA                SX1    X5                                                   CGIA        1510     A
CGIA                SA5    A5+B1                                                CGIA        1511     A
CGIA                SX6    X5                                                   CGIA        1512     A
CGIA                IX1    X6-X1                                                CGIA        1513     A
CGIA                CALL   COD=        LENGTH                                   CGIA        1514     A
CGIA                LX6    24                                                   CGIA        1515     A
CGIA                MX7    -12                                                  CGIA        1516     A
CGIA                BX6    X7*X6                                                CGIA        1517     A
CGIA                SA6    A6+B1                                                CGIA        1518     A
CGIA                LISTL  OBUF,8                                               CGIA        1519     A
CGIA                SX0    X0-1                                                 CGIA        1520     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                NZ     X0,USBLK18                                    CGIA        1521     A
CGIA                                                                     CGIA        1522     A
CGIA     *          LIST THE COMMON BLOCKS                                CGIA        1523     A
CGIA                                                                     CGIA        1524     A
CGIA                MOVE   3,BLKB,LBUF                                   CGIA        1525     A
CGIA                SA2    L.CBT                                         CGIA        1526     A
CGIA                SA3    O.CBT                                         CGIA        1527     A
CGIA                SX0    X2                                            CGIA        1528     A
CGIA                SA5    X3+B1                                         CGIA        1529     A
CGIA                ZR     X5,USBLK25  IF // COM NOT REFED               CGIA        1530     A
CGIA                                                                     CGIA        1531     A
CGIA      USBLK24   MX7    -24                                           CGIA        1532     A
CGIA                BX1    -X7*X5                                        CGIA        1533     A
CGIA                CALL   COD=                                          CGIA        1534     A
CGIA                LX6    24                                            CGIA        1535     A
CGIA                MX7    -12                                           CGIA        1536     A
CGIA                BX6    X7*X6                                         CGIA        1537     A
CGIA                SA6    LBUF+3                                        CGIA        1538     A
CGIA                LISTL  OBUF,8                                        CGIA        1539     A
CGIA                                                                     CGIA        1540     A
CGIA      USBLK25   SX0    X0-2        LEN = LEN - 2                     CGIA        1541     A
CGIA                ZR     X0,USBLK26  IF END OF CBT                     CGIA        1542     A
CGIA                SA1    A5+B1                                         CGIA        1543     A
CGIA                RJ     FSN         COUNT CHARS IN NAME               CGIA        1544     A
CGIA                SA3    =1H                                           CGIA        1545     A
CGIA                SB2    B3-B1                                         CGIA        1546     A
CGIA                MX2    B2                                            CGIA        1547     A
CGIA                BX3    -X2*X3                                        CGIA        1548     A
CGIA                BX6    X1+X3       BLANK FILL                        CGIA        1549     A
CGIA                SA6    LBUF                                          CGIA        1550     A
CGIA                SA5    A1+B1                                         CGIA        1551     A
CGIA                EQ     USBLK24                                       CGIA        1552     A
CGIA                                                                     CGIA        1553     A
CGIA      USBLK26   LISTL  (=8L ),1    BLANK LINE                        CGIA        1554     A
CGIA                                                                     CGIA        1555     A
CGIA     *          REFORMAT THE *PIDL* TABLE                            CGIA        1556     A
CGIA                                                                     CGIA        1557     A
CGIA      USBLK28   SA3    L.PIDL                                        CGIA        1558     A
CGIA                SA2    O.PIDL                                        CGIA        1559     A
CGIA                ZR     X3,USBLK30  IF L.PIDL = 0                     CGIA        1560     A
CGIA                BX0    -X3                                           CGIA        1561     A
CGIA                SA1    X2                                            CGIA        1562     A
CGIA                SX1    0           SET FOR BLANK COMMON              CGIA        1563     A
CGIA                                                                     CGIA        1564     A
CGIA      USBLK29   MX2    42                                            CGIA        1565     A
CGIA                BX1    X2*X1                                         CGIA        1566     A
CGIA                RJ     FSN         COUNT CHARS IN NAME               CGIA        1567     A
CGIA                SX3    1R/                                           CGIA        1568     A
CGIA                BX1    X1+X3                                         CGIA        1569     A
CGIA                LX1    -6                                            CGIA        1570     A
CGIA                SB4    B3-48                                         CGIA        1571     A
CGIA                LX3    -B4                                           CGIA        1572     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                BX1    X3+X1                                          CGIA        1573     A
CGIA                SX2    X2+12       NBITS = NBITS + 12                 CGIA        1574     A
CGIA                BX6    X1+X2                                          CGIA        1575     A
CGIA                SA6    A1                                             CGIA        1576     A
CGIA                SX0    X0+B1                                          CGIA        1577     A
CGIA                SA1    A1+B1                                          CGIA        1578     A
CGIA                NZ     X0,USBLK29                                     CGIA        1579     A
CGIA                                                                      CGIA        1580     A
CGIA     *          SETUP THE *ENTR* TABLE                                CGIA        1581     A
CGIA                                                                      CGIA        1582     A
CGIA      USBLK30   SA5    N.ENT                                          CGIA        1583     A
CGIA                ZR     X5,USBLK35  IF N.ENT = 0                       CGIA        1584     A
CGIA                LX4    B1,X5                                          CGIA        1585     A
CGIA                ALLOC  ENTR,X4+B1  ALLOC( ENTR , 2*N.ENT+1 )          CGIA        1586     A
CGIA                SB6    X2                                             CGIA        1587     A
CGIA                SB7    B1                                             CGIA        1588     A
CGIA                SB2    LT.ENTR-2000B                                  CGIA        1589     A
CGIA                LX5    37                                             CGIA        1590     A
CGIA                PX6    B2,X5                                          CGIA        1591     A
CGIA                SA6    B6          STORE HEADER WORD                  CGIA        1592     A
CGIA                LX5    -37                                            CGIA        1593     A
CGIA                SA4    O.SYM                                          CGIA        1594     A
CGIA                MX0    42                                             CGIA        1595     A
CGIA                SA1    X4+B1                                          CGIA        1596     A
CGIA                                                                      CGIA        1597     A
CGIA      USBLK31   SA1    A1+3                                           CGIA        1598     A
CGIA                LX1    59-WB.ENTP                                     CGIA        1599     A
CGIA                PL     X1,USBLK31  IF ^ENT[WORDB]                     CGIA        1600     A
CGIA                SA2    A1-B1                                          CGIA        1601     A
CGIA                SA3    A1+B1                                          CGIA        1602     A
CGIA                BX6    X0*X2                                          CGIA        1603     A
CGIA                SA6    B6+B7                                          CGIA        1604     A
CGIA                SX7    X3          RA                                 CGIA        1605     A
CGIA                SX4    B1          RB = 1        */ PROG RELOC        CGIA        1606     A
CGIA                LX3    58-AR.RLP                                      CGIA        1607     A
CGIA                PL     X3,USBLK32  IF ^ COMMON RELOC                  CGIA        1608     A
CGIA                LX3    2+AR.RLP-AR.RBP                                CGIA        1609     A
CGIA                MX4    -AR.RBL                                        CGIA        1610     A
CGIA                BX4    -X4*X3                                         CGIA        1611     A
CGIA                SA2    CCB                                            CGIA        1612     A
CGIA                IX4    X4+X2       RB = CCB + RB[WORDC]               CGIA        1613     A
CGIA      USBLK32   LX4    18                                             CGIA        1614     A
CGIA                BX7    X4+X7                                          CGIA        1615     A
CGIA                SA7    A6+B1                                          CGIA        1616     A
CGIA                SB7    B7+2                                           CGIA        1617     A
CGIA                SX5    X5-1                                           CGIA        1618     A
CGIA                NZ     X5,USBLK31                                     CGIA        1619     A
CGIA                                                                      CGIA        1620     A
CGIA                WRLGO  B6,B7       ENTRY TABLE TO LGO                 CGIA        1621     A
CGIA                                                                      CGIA        1622     A
CGIA     *          OUTPUT ENTRY STATEMENTS TO *COMPS*                    CGIA        1623     A
CGIA                                                                      CGIA        1624     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                SA1    BCDC                                          CGIA        1625     A
CGIA                PL     X1,USBLK34  IF C = 0                          CGIA        1626     A
CGIA                                                                     CGIA        1627     A
CGIA                SA1    O.ENTR                                        CGIA        1628     A
CGIA                SA2    L.ENTR                                        CGIA        1629     A
CGIA                AX0    B1,X2                                         CGIA        1630     A
CGIA                SA5    X1+B1                                         CGIA        1631     A
CGIA                BX3    X5                                            CGIA        1632     A
CGIA                                                                     CGIA        1633     A
CGIA       USBLK33  SA2    =8LENTRY                                      CGIA        1634     A
CGIA                RJ     FSP                                           CGIA        1635     A
CGIA                SX0    X0-1                                          CGIA        1636     A
CGIA                SA5    A5+2                                          CGIA        1637     A
CGIA                BX3    X5                                            CGIA        1638     A
CGIA                NZ     X0,USBLK33                                    CGIA        1639     A
CGIA                                                                     CGIA        1640     A
CGIA       USBLK34  SX6    0                                             CGIA        1641     A
CGIA                SA6    L.ENTR      L.ENTR = 0                        CGIA        1642     A
CGIA                                                                     CGIA        1643     A
CGIA       USBLK35  SA1    BCDC                                          CGIA        1644     A
CGIA                ZR     X1,USBLK45  IF BCDC = 0                       CGIA        1645     A
CGIA                SA2    XCI                                           CGIA        1646     A
CGIA                LX2    -AR.XCIP                                      CGIA        1647     A
CGIA                SX5    X2                                            CGIA        1648     A
CGIA                ZR     X5,USBLK45  IF XCI = 0                        CGIA        1649     A
CGIA                PL     X1,USBLK38  IF OL                             CGIA        1650     A
CGIA                                                                     CGIA        1651     A
CGIA     *          OUTPUT *EXT* STATEMENTS TO *COMPS*                   CGIA        1652     A
CGIA                                                                     CGIA        1653     A
CGIA       USBLK36  SA2    =8LEXT                                        CGIA        1654     A
CGIA                SA4    O.SYM                                         CGIA        1655     A
CGIA                SB5    X5                                            CGIA        1656     A
CGIA                SA3    X4+B5       WORD A                            CGIA        1657     A
CGIA                SA5    A3+2        WORD C                            CGIA        1658     A
CGIA                MX7    42                                            CGIA        1659     A
CGIA                BX3    X7*X3                                         CGIA        1660     A
CGIA                LX5    -AR.XCIP                                      CGIA        1661     A
CGIA                SX5    X5                                            CGIA        1662     A
CGIA                RJ     FSP         OUTPUT EXT STMT                   CGIA        1663     A
CGIA                NZ     X5,USBLK36  IF NOT END OF CHAIN               CGIA        1664     A
CGIA                                                                     CGIA        1665     A
CGIA                EQ     USBLK45                                       CGIA        1666     A
CGIA                                                                     CGIA        1667     A
CGIA     *          OUTPUT O-LIST OF EXTERNAL NAMES                      CGIA        1668     A
CGIA                                                                     CGIA        1669     A
CGIA       USBLK38  SA1    =10HEXTERNALS                                 CGIA        1670     A
CGIA                MX7    0                                             CGIA        1671     A
CGIA                BX6    X1                                            CGIA        1672     A
CGIA                SA6    LBUF                                          CGIA        1673     A
CGIA                SA7    A6+B1                                         CGIA        1674     A
CGIA                LISTL  OBUF,6                                        CGIA        1675     A
CGIA                                                                     CGIA        1676     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA      USBLK39  SB7    LBUF        INDEX                                  CGIA        1677     A
CGIA               SB6    LBUF+6      LIMIT                                  CGIA        1678     A
CGIA               SA1    O.SYM                                              CGIA        1679     A
CGIA               SB5    X1+2                                               CGIA        1680     A
CGIA                                                                         CGIA        1681     A
CGIA      USBLK40  SA4    B5+X5                                              CGIA        1682     A
CGIA               LX4    -AR.XCIP                                           CGIA        1683     A
CGIA               SX5    X4          NEXT                                   CGIA        1684     A
CGIA               SA1    A4-2                                               CGIA        1685     A
CGIA               MX2    -6                                                 CGIA        1686     A
CGIA               BX6    X2*X1                                              CGIA        1687     A
CGIA               BX1    -X2*X1                                             CGIA        1688     A
CGIA               SB3    X1                                                 CGIA        1689     A
CGIA               SA3    =1H                                                CGIA        1690     A
CGIA               SB2    B3-B1                                              CGIA        1691     A
CGIA               MX2    B2                                                 CGIA        1692     A
CGIA               BX3    -X2*X3                                             CGIA        1693     A
CGIA               BX6    X6+X3                                              CGIA        1694     A
CGIA               SA6    B7                                                 CGIA        1695     A
CGIA               SB7    B7+B1                                              CGIA        1696     A
CGIA               ZR     X5,USBLK41  IF END OF THE CHAIN                    CGIA        1697     A
CGIA               LT     B7,B6,USBLK40     IF LINE NOT FULL                 CGIA        1698     A
CGIA                                                                         CGIA        1699     A
CGIA      USBLK41  MX6    0                                                  CGIA        1700     A
CGIA               SA6    B7                                                 CGIA        1701     A
CGIA               SB7    B7+B1                                              CGIA        1702     A
CGIA               LISTL  OBUF,B7-B6                                         CGIA        1703     A
CGIA               NZ     X5,USBLK39  IF MORE NAMES TO PROCESS               CGIA        1704     A
CGIA                                                                         CGIA        1705     A
CGIA               LISTL  (=8L ),1                                           CGIA        1706     A
CGIA                                                                         CGIA        1707     A
CGIA     *         REFORMAT *CBT* FOR USE BY CUB, BIN OUTPUT ROUTINES, ETC   CGIA        1708     A
CGIA                                                                         CGIA        1709     A
CGIA      USBLK45  SA1    O.LBT                                              CGIA        1710          I
          -CCG14
CGIA      USBLK45  SA1    O.CBT                                              CCG14          6     A
CGIA               SA2    L.CBT                                              CGIA        1711     A
CGIA               SA3    CCB                                                CGIA        1712     A
CGIA               SB5    AD.RBP                                             CGIA        1713     A
CGIA               SB6    4                                                  CGIA        1714     A
CGIA               MX0    1                                                  CGIA        1715     A
CGIA               SB2    X1          I = 0                                  CGIA        1716     A
CGIA               SB3    X2+B2       N = L.CBT                              CGIA        1717     A
CGIA               BX7    X3          R = CCB                                CGIA        1718     A
CGIA               SB4    2                                                  CGIA        1719     A
CGIA                                                                         CGIA        1720     A
CGIA      USBLK46  SA2    B2+B1                                              CGIA        1721     A
CGIA               BX4    X0*X2                                              CGIA        1722     A
CGIA               SA7    A2          CBT(I+1) = R                           CGIA        1723     A
CGIA               LX4    1+AD.LCMP                                          CGIA        1724     A
CGIA               LX6    B5,X7                                              CGIA        1725     A
CGIA               PX5    B6,X6                                              CGIA        1726     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA            BX6    X4+X5                                                CGIA      1727    A
CGIA            SA6    B2                                                   CGIA      1728    A
CGIA            SX7    X7+B1                                                CGIA      1729    A
CGIA            SB2    B2+B4       I = I + 2                                CGIA      1730    A
CGIA            LT     B2,B3,USBLK46                                        CGIA      1731    A
CGIA                                                                        CGIA      1732    A
CGIA     *      SETUP TEXT TABLE, ETC FOR PROCESSING                        CGIA      1733    A
CGIA                                                                        CGIA      1734    A
CGIA            SA1    IL.LAF                                               CGIA      1735    A
CGIA            SX7    0                                                    CGIA      1736    A
CGIA            SA7    L.LAFT      L.LAFT = 0                               CGIA      1737    A
CGIA            ALLOC  LAFT,X1     ALLOC( LAFT , IL.LAF )  */ INITIAL ALLOC CGIA      1738    A
CGIA            IFEQ   HC.TS,0,2                                            CGIA      1739    A
CGIA            SX6    =XTOVA#                                              CGIA      1740    A
CGIA            SA6    =XTO#       TO = TOVA     */ SET ASSEMBLER OVERFLOW EXIT CGIA 1741    A
CGIA            SX0    B1                                                   CGIA      1742    A
CGIA            LX0    AD.RBP                                               CGIA      1743    A
CGIA            SB2    4           PR = 4                                   CGIA      1744    A
CGIA            MX6    0                                                    CGIA      1745    A
CGIA            PX7    B2,X0                                                CGIA      1746    A
CGIA            SA7    AI                                                   CGIA      1747    A
CGIA            SA6    CW          CW = 0                                   CGIA      1748    A
CGIA            SA6    TI          TI = 0                                   CGIA      1749    A
CGIA            SB3    LT.XTXT-2000B                                        CGIA      1750    A
CGIA            PX7    B3,X0                                                CGIA      1751    A
CGIA            SA7    A6+B1       TH = 12/LT.XTXT,12/0,36/RBOC             CGIA      1752    A
CGIA            SA6    A7+B1       TR = 0                                   CGIA      1753    A
CGIA            SA6    RBWORD      RBWORD = 0                               CGIA      1754    A
CGIA            SA4    IBUF                                                 CGIA      1755    A
CGIA            SA5    X4          II = [IBUF]   */ RESTORE INPUT BUFFER PTR CGIA     1756    A
CGIA            EQ     GNIW                                                 CGIA      1757    A
CGIA                                                                        CGIA      1758    A
CGIA      BLKA  DIS    ,/BLOCK     TYPE      ADDRESS   LENGTH/              CGIA      1759    A
CGIA      BLKB  DATA   30H/ /       COMMON         0                        CGIA      1760    A
CGIA      FSN   SPACE  2,14                                                 CGIA      1761    A
CGIA     **     FSN - FORMAT SYMBOL NAME                                   CGIA      1762    A
CGIA     *                                                                  CGIA      1763    A
CGIA     *      ENTRY  (X1) = 0L_NAME                                      CGIA      1764    A
CGIA     *                                                                  CGIA      1765    A
CGIA     *      EXIT   (X6) = 54/0L_NAME,6/NBITS                           CGIA      1766    A
CGIA     *             (X2,B3) = NBITS , 6*N-CHARS-IN-NAME                 CGIA      1767    A
CGIA     *                                                                  CGIA      1768    A
CGIA     *      USES   X1 - X7, B3, A5                                     CGIA      1769    A
CGIA                                                                        CGIA      1770    A
CGIA      FSN   ROUTINE                                                     CGIA      1771    A
CGIA            SX3    B1                                                   CGIA      1772    A
CGIA            SA5    =4040404040404040404040B                             CGIA      1773    A
CGIA            IX4    X1-X3                                                CGIA      1774    A
CGIA            BX3    -X4+X1                                               CGIA      1775    A
CGIA            SB3    60-5                                                 CGIA      1776    A
CGIA            BX4    X5*X3                                                CGIA      1777    A
CGIA            LX3    B3,X4                                                CGIA      1778    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA              IX2    X4-X3                                          CGIA        1779     A
CGIA              BX6    X2+X4       MASK OF 77 "S                      CGIA        1780     A
CGIA              AX6    12                                             CGIA        1781     A
CGIA              BX4    -X6                                            CGIA        1782     A
CGIA              PX6    X4                                             CGIA        1783     A
CGIA              NX3    B3,X6                                          CGIA        1784     A
CGIA              SX2    B3                                             CGIA        1785     A
CGIA              BX6    X1+X2                                          CGIA        1786     A
CGIA              EQ     FSN                                            CGIA        1787     A
CGIA       FSP    SPACE  3,14                                           CGIA        1788     A
CGIA     **       FSP - FORMAT AND LIST SIMPLE PSEUDO                   CGIA        1789     A
CGIA     *                                                              CGIA        1790     A
CGIA     *        ENTRY  (X2) = 8L_PSEUDO-NAME                          CGIA        1791     A
CGIA     *               (X3) = 0LNAME                                  CGIA        1792     A
CGIA                                                                    CGIA        1793     A
CGIA       FSP    ROUTINE                                               CGIA        1794     A
CGIA              SA1    =1H                                            CGIA        1795     A
CGIA              LX3    12                                             CGIA        1796     A
CGIA              BX7    X1                                             CGIA        1797     A
CGIA              MX4    48                                             CGIA        1798     A
CGIA              SA7    LBUF                                           CGIA        1799     A
CGIA              BX6    -X4*X3                                         CGIA        1800     A
CGIA              BX7    X2+X6                                          CGIA        1801     A
CGIA              SA7    A7+B1                                          CGIA        1802     A
CGIA              BX7    X4*X3                                          CGIA        1803     A
CGIA              SB4    12                                             CGIA        1804     A
CGIA              RJ     TSB         TERMINATE BUFFER                   CGIA        1805     A
CGIA              WRTEXT 0           AND LIST IT                        CGIA        1806     A
CGIA              EQ     FSP                                            CGIA        1807     A
CGIA       CG$ASN SPACE  3,14                                           CGIA        1808     A
CGIA     **       CG$ASN - ADJUST SYMBOL NAMES IN *SYM* AND *QBT*       CGIA        1809     A
CGIA                                                                    CGIA        1810     A
CGIA       CG$ASN ENTRY. **                                             CGIA        1811     A
CGIA                                                                    CGIA        1812     A
CGIA     *        REFORMAT NAME WORDS OF SYMTAB                         CGIA        1813     A
CGIA                                                                    CGIA        1814     A
CGIA              SA1    O.SYM                                          CGIA        1815     A
CGIA              SA2    L.SYM                                          CGIA        1816     A
CGIA              SB5    X2-3                                           CGIA        1817     A
CGIA              SA1    X1+3                                           CGIA        1818     A
CGIA                                                                    CGIA        1819     A
CGIA       ASN1   MX0    6*HC.MCIS                                      CGIA        1820     A
CGIA              BX1    X0*X1                                          CGIA        1821     A
CGIA              RJ     FSN         FORMAT NAME                        CGIA        1822     A
CGIA              SA6    A1                                             CGIA        1823     A
CGIA              SB5    B5-3                                           CGIA        1824     A
CGIA              SA1    A1+3                                           CGIA        1825     A
CGIA              NZ     B5,ASN1                                        CGIA        1826     A
CGIA                                                                    CGIA        1827     A
CGIA     *        REFORMAT *QBT*                                        CGIA        1828     A
CGIA                                                                    CGIA        1829     A
CGIA       .QBT   IFNE   HC.QBT,0                                       CGIA        1830     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA              SA1    O.QBT                                                  CGIA        1831     A
CGIA              SA2    L.QBT                                                  CGIA        1832     A
CGIA              SB5    X2-2                                                   CGIA        1833     A
CGIA              SA1    X1+B1                                                  CGIA        1834     A
CGIA              LE     B5,ASN3                                                CGIA        1835     A
CGIA                                                                            CGIA        1836     A
CGIA      ASN2    SA1    A1+1                                                   CGIA        1837     A
CGIA              RJ     FSN                                                    CGIA        1838     A
CGIA              SA6    A1                                                     CGIA        1839     A
CGIA              SB5    B5-B1                                                  CGIA        1840     A
CGIA              NZ     B5,ASN2                                                CGIA        1841     A
CGIA                                                                            CGIA        1842     A
CGIA      ASN3    BSS    0                                                      CGIA        1843     A
CGIA                                                                            CGIA        1844     A
CGIA      .QBT    ENDIF                                                          CGIA        1845     A
CGIA                                                                            CGIA        1846     A
CGIA     *        SETUP *EXT* CHAIN AND COUNT NUMBER OF ENTRY POINTS AND         CGIA        1847     A
CGIA     *        EXTERNALS                                                      CGIA        1848     A
CGIA                                                                            CGIA        1849     A
CGIA              SA1    O.SYM                                                  CGIA        1850     A
CGIA              SA2    L.SYM                                                  CGIA        1851     A
CGIA              SA5    ASNA                                                   CGIA        1852     A
CGIA              SB4    X1+2                                                   CGIA        1853     A
CGIA              SB5    X2-3                                                   CGIA        1854     A
CGIA              SB6    B0          N.ENT = 0                                  CGIA        1855     A
CGIA              SB7    B0          N.EXT = 0                                  CGIA        1856     A
CGIA              SA1    X1+B1                                                  CGIA        1857     A
CGIA              MX7    2           XC = 0                                     CGIA        1858     A
CGIA              EQ     ASN5                                                   CGIA        1859     A
CGIA                                                                            CGIA        1860     A
CGIA      ASN4    SB6    B6+1        N.ENT = N.ENT + 1                          CGIA        1861     A
CGIA                                                                            CGIA        1862     A
CGIA      ASN5    SA1    A1+3                                                   CGIA        1863     A
CGIA              SB5    B5-3        LS = LS - 3                                CGIA        1864     A
CGIA              MI     B5,ASN6     IF END OF SYMTAB                           CGIA        1865     A
CGIA              BX2    X1                                                     CGIA        1866     A
CGIA              LX2    59-WB.LABP                                             CGIA        1867     A
CGIA              MI     X2,ASN5     IF LAB[WORDB]                              CGIA        1868     A
CGIA              BX4    X5*X1                                                  CGIA        1869     A
CGIA              LX1    59-WB.ENTP                                             CGIA        1870     A
CGIA              ZR     X4,ASN5     IF ^( EXT ! ENT )                          CGIA        1871     A
CGIA              MI     X1,ASN4     IF ENT                                     CGIA        1872     A
CGIA                                                                            CGIA        1873     A
CGIA              SB7    B7+B1       N.EXT = N.EXT + 1                          CGIA        1874     A
CGIA              SA7    A1+B1       WORDC = XC    */ CHAIN POINTS TO LAST      CGIA        1875     A
CGIA              SX4    A7-B4                                                  CGIA        1876     A
CGIA              MX6    2                                                      CGIA        1877     A
CGIA              LX4    AR.XCIP                                                CGIA        1878     A
CGIA              BX7    X6+X4       XCI = INDEX OF THIS                        CGIA        1879     A
CGIA              EQ     ASN5                                                   CGIA        1880     A
CGIA                                                                            CGIA        1881     A
CGIA      ASN6    SA7    XCI                                                    CGIA        1882     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA              SX6    B6                                                  CGIA        1883     A
CGIA              SX7    B7+B1                                               CGIA        1884     A
CGIA              SA6    N.ENT                                               CGIA        1885     A
CGIA              AX7    1                                                   CGIA        1886     A
CGIA              SX7    X7+B1                                               CGIA        1887     A
CGIA              SA7    IL.LAF      IL.LAF = (N.EXT+1)/2 + 1                CGIA        1888     A
CGIA              EQ     CG$ASN                                              CGIA        1889     A
CGIA                                                                        CGIA        1890     A
CGIA      ASNA    BFMW   WB,(EXT,ENT)                                       CGIA        1891     A
CGIA      ADC     TITLE  DISPLAY CODE CONVERSION ROUTINES                   CGIA        1892     A
CGIA     **       BFN - BLANK FILL NAME MACRO                               CGIA        1893     A
CGIA     *                                                                  CGIA        1894     A
CGIA     *        ENTRY  X6,B3 = NAME AND BIT COUNT                         CGIA        1895     A
CGIA     *        EXIT   BLANK FILLED NAME IN REGISTER *RR*                 CGIA        1896     A
CGIA     *                                                                  CGIA        1897     A
CGIA     *        USES   A2, B2, X2, X3                                     CGIA        1898     A
CGIA                                                                        CGIA        1899     A
CGIA      BFN     MACRO  RR                                                 CGIA        1900     A
CGIA              SA3    =1H                                                 CGIA        1901     A
CGIA              SB2    B3-B1                                               CGIA        1902     A
CGIA              MX2    B2                                                  CGIA        1903     A
CGIA              BX3    -X2*X3                                              CGIA        1904     A
CGIA              B;A    X6+X3                                               CGIA        1905     A
CGIA              ENDM                                                       CGIA        1906     A
CGIA      ADC     SPACE  2                                                   CGIA        1907     A
CGIA     **       ADC - ADD CHARACTERS TO STRING                            CGIA        1908     A
CGIA     *                                                                  CGIA        1909     A
CGIA     *        ENTRY  (X6) = CHAR STRING                                 CGIA        1910     A
CGIA     *               (B3) = BC , BIT COUNT                              CGIA        1911     A
CGIA     *               (X7) = ASSEMBLED STRING IN 0L FORMAT               CGIA        1912     A
CGIA     *               (B4) = UBC , UNUSED BIT COUNT ( FOR X7 )           CGIA        1913     A
CGIA     *                                                                  CGIA        1914     A
CGIA     *        USES   X - 2, 3, 4, 6   B - 2, 3, 4                       CGIA        1915     A
CGIA                                                                        CGIA        1916     A
CGIA      ADC     ROUTINE                                                   CGIA        1917     A
CGIA              SB2    B4-B1                                               CGIA        1918     A
CGIA              MX4    B2                                                  CGIA        1919     A
CGIA              BX3    X4*X6                                               CGIA        1920     A
CGIA              LX2    B4,X3                                               CGIA        1921     A
CGIA              BX7    X2+X7                                               CGIA        1922     A
CGIA              BX4    -X4*X6                                              CGIA        1923     A
CGIA              LX6    B4,X4                                               CGIA        1924     A
CGIA              SB4    B4-B3       UBC = UBC - BC                          CGIA        1925     A
CGIA              GT     B4,ADC      IF UBC > 0                              CGIA        1926     A
CGIA              SA7    A7+B1       STORE CURRENT WORD                      CGIA        1927     A
CGIA              SB4    B4+60       UBC = UBC + 60                          CGIA        1928     A
CGIA              BX7    X6                                                  CGIA        1929     A
CGIA              EQ     ADC                                                 CGIA        1930     A
CGIA      TSB     SPACE  2                                                   CGIA        1931     A
CGIA     **       TSB - TERMINATE STRING BUFFER                             CGIA        1932     A
CGIA     *                                                                  CGIA        1933     A
CGIA     *        ENTRY  (X7) LAST WORD OF ASSEMBLED STRING                 CGIA        1934     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA     *                (B4) = UBC                                                CGIA     1935     A
CGIA     *                                                                          CGIA     1936     A
CGIA     *         EXIT   LINE WRITTEN TO *COMPS* IF BCDC < 0                        CGIA     1937     A
CGIA     *                (LBUF.WC) = NUMBER OF WORDS IN LBUF                       CGIA     1938     A
CGIA     *                                                                          CGIA     1939     A
CGIA     *         PRESERVES X1 IF C = 0 ( NEEDED IN *RJXJ* PROCESSOR )              CGIA     1940     A
CGIA                                                                                CGIA     1941     A
CGIA      TSB0     SB7    A7+B1                                                     CGIA     1942     A
CGIA               SB6    LBUF                                                      CGIA     1943     A
CGIA               SB7    B7-B6       (B7) = WORD COUNT                             CGIA     1944     A
CGIA               SA2    BCDC                                                      CGIA     1945     A
CGIA               SX6    B7+                                                       CGIA     1946     A
CGIA               SA6    LBUF.WC                                                   CGIA     1947     A
CGIA               PL     X2,TSB      IF BCDC \ 0                                   CGIA     1948     A
CGIA               WRITEC F.CMPS,B6,B7      WRITE LINE TO COMPS FILE                 CGIA     1949     A
CGIA                                                                                CGIA     1950     A
CGIA      TSB      ROUTINE                                                           CGIA     1951     A
CGIA               SA7    A7+1        STORE LAST WORD                               CGIA     1952     A
CGIA               SB3    12                                                        CGIA     1953     A
CGIA               GE     B4,B3,TSB0  IF 12 BIT LINE TERMINATOR PRESENT             CGIA     1954     A
CGIA               MX7    0                                                         CGIA     1955     A
CGIA               SA7    A7+B1       STORE A ZERO WORD                             CGIA     1956     A
CGIA               EQ     TSB0                                                      CGIA     1957     A
CGIA      CMA      SPACE  3,14                                                      CGIA     1958     A
CGIA     **        CMA - CONVERT MACRO ARGUMENT TO DISPLAY CODE                      CGIA     1959     A
CGIA     *                                                                          CGIA     1960     A
CGIA     *         ENTRY  (X2) = ARG VALUE                                           CGIA     1961     A
CGIA     *                (X4) = ARG TYPE ( 1 - 6 )                                 CGIA     1962     A
CGIA     *                                                                          CGIA     1963     A
CGIA     *         EXIT   (X6) = 0L_BCD-STRING                                       CGIA     1964     A
CGIA     *                (B3) = BC , BIT COUNT ( 6*N.CHARS IN STRING )             CGIA     1965     A
CGIA     *                                                                          CGIA     1966     A
CGIA     *         USES   X - 2, 3, 4, 6   B - 2, 3                                  CGIA     1967     A
CGIA                                                                                CGIA     1968     A
CGIA      CMA      ROUTINE                                                           CGIA     1969     A
CGIA               SB2    X4                                                        CGIA     1970     A
CGIA               JP     CMA.JT-1+B2                                               CGIA     1971     A
CGIA                                                                                CGIA     1972     A
CGIA      CMA.JT   BX6    X2                                                        CGIA     1973     A
CGIA               LX2    59-23                                                     CGIA     1974     A
CGIA               EQ     CMA.C                                                     CGIA     1975     A
CGIA                                                                                CGIA     1976     A
CGIA               EQ     CMA.B                                                     CGIA     1977     A
CGIA                                                                                CGIA     1978     A
CGIA               EQ     CMA.S                                                     CGIA     1979     A
CGIA                                                                                CGIA     1980     A
CGIA               SB2    X2                                                        CGIA     1981     A
CGIA               MX6    -6                                                        CGIA     1982     A
CGIA               EQ     CMA.Q                                                     CGIA     1983     A
CGIA                                                                                CGIA     1984     A
CGIA               SX6    1R(                                                       CGIA     1985     A
CGIA               EQ     CMA.M                                                     CGIA     1986     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                                                                                    CGIA        1987     A
CGIA     *         CONVERT CONSTANT                                                     CGIA        1988     A
CGIA                                                                                    CGIA        1989     A
CGIA      CMA.C    MI      X2,CMA.C1    IF VAL < 0                                      CGIA        1990     A
CGIA               RJ      CON                                                          CGIA        1991     A
CGIA               LX6     6                                                            CGIA        1992     A
CGIA               SB3     B3-6                                                         CGIA        1993     A
CGIA               EQ      CMA                                                          CGIA        1994     A
CGIA                                                                                    CGIA        1995     A
CGIA      CMA.C1   AX2     59-23                                                        CGIA        1996     A
CGIA               BX6     -X2                                                          CGIA        1997     A
CGIA               RJ      CON                                                          CGIA        1998     A
CGIA               SX4     1R-                                                          CGIA        1999     A
CGIA               LX4     54                                                           CGIA        2000     A
CGIA               BX6     X4+X6        *-NNNB*                                         CGIA        2001     A
CGIA               EQ      CMA                                                          CGIA        2002     A
CGIA                                                                                    CGIA        2003     A
CGIA      CMA.S    RJ      CSN          CONVERT NAME                                    CGIA        2004          I
          -CCG16
CGIA      CMA.S    CALL    CSN#         CONVERT NAME                                    CCG16         45     A
CGIA               EQ      CMA                                                          CGIA        2005     A
CGIA                                                                                    CGIA        2006     A
CGIA      CMA.Q    BSS     0                                                            CGIA        2007     A
CGIA      .HC      IFNE    HC.QBT,0                                                     CGIA        2008     A
CGIA               SA3     O.QBT        QUAL BLOCK NAME                                 CGIA        2009     A
CGIA               SA4     X3+B2                                                        CGIA        2010     A
CGIA               BX3     -X6*X4                                                       CGIA        2011     A
CGIA               SB3     X3                                                           CGIA        2012     A
CGIA               BX6     X6*X4                                                        CGIA        2013     A
CGIA               EQ      CMA                                                          CGIA        2014     A
CGIA      .HC      ENDIF                                                                CGIA        2015     A
CGIA                                                                                    CGIA        2016     A
CGIA      CMA.B    SX6     X2-1S15                                                      CGIA        2017     A
CGIA               MI      X6,CMA.B1    IF A COMMON BLOCK                               CGIA        2018     A
CGIA               SA3     LBNT+X6                                                      CGIA        2019     A
CGIA               SB3     42                                                           CGIA        2020     A
CGIA               MX6     42                                                           CGIA        2021     A
CGIA               BX6     X6*X3                                                        CGIA        2022     A
CGIA               EQ      CMA                                                          CGIA        2023     A
CGIA                                                                                    CGIA        2024     A
CGIA      CMA.B1   SA3     O.PIDL                                                       CGIA        2025     A
CGIA               MX6     -6                                                           CGIA        2026     A
CGIA               SB2     X2                                                           CGIA        2027     A
CGIA               SA4     X3+B2                                                        CGIA        2028     A
CGIA               BX3     -X6*X4                                                       CGIA        2029     A
CGIA               SB3     X3                                                           CGIA        2030     A
CGIA               BX6     X6*X4                                                        CGIA        2031     A
CGIA               EQ      CMA                                                          CGIA        2032     A
CGIA                                                                                    CGIA        2033     A
CGIA      CMA.M    LX2     1            CONVERT MICRO NUMBER TO MIC VALUE               CGIA        2034     A
CGIA               SB6     X2                                                           CGIA        2035     A
CGIA               SB3     6                                                            CGIA        2036     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA            LX6    -6                                                  CGIA      2037    A
CGIA            RJ     ADC         ADD LEFT PAREN                          CGIA      2038    A
CGIA            SA4    =XF$MIC-2+B6                                        CGIA      2039    A
CGIA            SA3    A4+B1                                               CGIA      2040    A
CGIA            BX6    X4                                                  CGIA      2041    A
CGIA            SB3    X3                                                  CGIA      2042    A
CGIA            RJ     ADC         ADD STRING                              CGIA      2043    A
CGIA            SX6    1R)                                                 CGIA      2044    A
CGIA            SB3    6                                                   CGIA      2045    A
CGIA            LX6    -6                                                  CGIA      2046    A
CGIA            EQ     CMA                                                 CGIA      2047    A
CGIA      CON   SPACE  3,14                                                CGIA      2048    A
CGIA     **     CON - CONVERT OCTAL NUMBER TO DISPLAY CODE                 CGIA      2049    A
CGIA     *                                                                 CGIA      2050    A
CGIA     *      ENTRY  (X6) = BINARY NUMBER ( \ 0 )                        CGIA      2051    A
CGIA     *                                                                 CGIA      2052    A
CGIA     *      EXIT   (X6) = 0L:NNNB                                      CGIA      2053    A
CGIA     *             (B3) = BIT COUNT                                    CGIA      2054    A
CGIA     *                                                                 CGIA      2055    A
CGIA     *      USES   X - 2, 3, 4, 6   B - 2, 3                           CGIA      2056    A
CGIA                                                                      CGIA      2057    A
CGIA      CON0  SX6    X6+1R0                                              CGIA      2058    A
CGIA            LX6    -12                                                 CGIA      2059    A
CGIA      CON   ROUTINE                                                    CGIA      2060    A
CGIA            MX2    -3                                                  CGIA      2061    A
CGIA            SB3    12          BC = 12                                 CGIA      2062    A
CGIA            BX3    X2*X6                                               CGIA      2063    A
CGIA            ZR     X3,CON0     IF NUM < 8                              CGIA      2064    A
CGIA            SX4    1RB                                                 CGIA      2065    A
CGIA                                                                      CGIA      2066    A
CGIA      CON1  BX3    -X2*X6                                              CGIA      2067    A
CGIA            LX4    -6                                                  CGIA      2068    A
CGIA            SX3    X3+1R0                                              CGIA      2069    A
CGIA            AX6    3                                                   CGIA      2070    A
CGIA            IX4    X3+X4                                               CGIA      2071    A
CGIA            SB3    B3+6        BC = BC + 6                             CGIA      2072    A
CGIA            NZ     X6,CON1                                             CGIA      2073    A
CGIA                                                                      CGIA      2074    A
CGIA            LX4    -12                                                 CGIA      2075    A
CGIA            BX6    X4                                                  CGIA      2076    A
CGIA            EQ     CON                                                 CGIA      2077    A
CGIA      CSN   SPACE  3,14                                                CGIA      2078          I
         -CCG16
CGIA     **     CSN - CONVERT SYMTAB *IH* TO BCD NAME                      CGIA      2079          I
         -CCG16
CGIA     *                                                                 CGIA      2080          I
         -CCG16
CGIA     *      ENTRY  (X2) = *IH*                                         CGIA      2081          I
         -CCG16
CGIA     *                                                                 CGIA      2082          I
         -CCG16
CGIA     *      EXIT   (X6) = 0L_BCD-NAME                                  CGIA      2083          I

         0        1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
                -CCG16
CGIA            *                (B3) = BC                                                  CGIA        2084          I
                -CCG16
CGIA            *                                                                           CGIA        2085          I
                -CCG16
CGIA            *         USES   X - 2, 3, 4, 6   B - 2, 3                                  CGIA        2086          I
                -CCG16
CGIA                                                                                        CGIA        2087          I
                -CCG16
CGIA             CSN0     LX3    B1,X2                                                      CGIA        2088          I
                -CCG16
CGIA                      SB2    X4                                                         CGIA        2089          I
                -CCG16
CGIA                      IX3    X2+X3                                                      CGIA        2090          I
                -CCG16
CGIA                      SA4    B2+X3       WORDA = SYM(IH)                                CGIA        2091          I
                -CCG16
CGIA                      MX6    -6                                                         CGIA        2092          I
                -CCG16
CGIA                      BX3    -X6*X4                                                     CGIA        2093          I
                -CCG16
CGIA                      SB3    X3                                                         CGIA        2094          I
                -CCG16
CGIA                      BX6    X6*X4                                                      CGIA        2095          I
                -CCG16
CGIA                                                                                        CGIA        2096          I
                -CCG16
CGIA             CSN      ROUTINE                                                            CGIA        2097          I
                -CCG16
CGIA                      MX3    -IH.HL                                                     CGIA        2098          I
                -CCG16
CGIA                      BX6    X3*X2                                                      CGIA        2099          I
                -CCG16
CGIA                      SA4    O.SYM                                                      CGIA        2100          I
                -CCG16
CGIA                      ZR     X6,CSN0     IF I[IH] = 0                                   CGIA        2101          I
                -CCG16
CGIA                                                                                        CGIA        2102          I
                -CCG16
CGIA            *         CONVERT FOWARD REFERENCE SYMBOL TO DISPLAY CODE.                  CGIA        2103          I
                -CCG16
CGIA                                                                                        CGIA        2104          I
                -CCG16
CGIA                      BX4    -X3*X2                                                     CGIA        2105          I
                -CCG16
CGIA                      AX2    IH.HL                                                      CGIA        2106          I
                -CCG16
CGIA                      SA3    CSNA-1+X2                                                  CGIA        2107          I
                -CCG16
CGIA                      LX4    60-IH.HL                                                   CGIA        2108          I
                -CCG16
CGIA                      BX6    X3                                                         CGIA        2109          I

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
                -CCG16
CGIA                    SB3     9*6-1       BC = 54-1                                   CGIA        2110        I
                -CCG16
CGIA                                                                                    CGIA        2111        I
                -CCG16
CGIA            +       LX4     3                                                       CGIA        2112        I
                -CCG16
CGIA                    SB3     B3-6        BC = BC - 6                                 CGIA        2113        I
                -CCG16
CGIA                    SX3     X4                                                      CGIA        2114        I
                -CCG16
CGIA                    ZR      X3,*-1      SKIP PAST LEADING ZEROS                     CGIA        2115        I
                -CCG16
CGIA                                                                                    CGIA        2116        I
                -CCG16
CGIA                    SB2     60-4*6                                                  CGIA        2117        I
                -CCG16
CGIA             CSN1   IX4     X4-X3       REMOVE DIGIT                                CGIA        2118        I
                -CCG16
CGIA                    LX3     B2                                                      CGIA        2119        I
                -CCG16
CGIA                    IX6     X6+X3       ADD DIGIT TO STRING                         CGIA        2120        I
                -CCG16
CGIA                    SB2     B2-6                                                    CGIA        2121        I
                -CCG16
CGIA                    LX4     3                                                       CGIA        2122        I
                -CCG16
CGIA                    SX3     X4                                                      CGIA        2123        I
                -CCG16
CGIA                    NZ      X4,CSN1                                                 CGIA        2124        I
                -CCG16
CGIA                                                                                    CGIA        2125        I
                -CCG16
CGIA                    MX3     1                                                       CGIA        2126        I
                -CCG16
CGIA                    AX3     B3                                                      CGIA        2127        I
                -CCG16
CGIA                    BX6     X3*X6       REMOVE GARBAGE FROM THE NAME                CGIA        2128        I
                -CCG16
CGIA                    SB3     B3+B1       BC = BC + 1                                 CGIA        2129        I
                -CCG16
CGIA                    EQ      CSN                                                     CGIA        2130        I
                -CCG16
CGIA                                                                                    CGIA        2131        I
                -CCG16
CGIA             CSNA   BSS     0                                                       CGIA        2132        I
                -CCG16
CGIA                    ECHO    1,Z=("HC.FRTP")                                         CGIA        2133        I
                -CCG16
CGIA                    DATA    8L;A.00000                                              CGIA        2134        I
                -CCG16
CGIA             FMC    SPACE   3,14                                                    CGIA        2135    A

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA     **        FMC - FORMAT MACRO CALL ( CONVERT TO BCD )                          CGIA      2136    A
CGIA     *                                                                             CGIA      2137    A
CGIA     *         ENTRY  (X1) = MACRO HEADER WORD ( 36/6L_NAME,3/LAB,18/PAR,3/NP      CGIA      2138    A
CGIA     *                                                                             CGIA      2139    A
CGIA     *         EXIT   MACRO CALL FORMATTED AS A BCD LINE IMAGE AND STORED IN       CGIA      2140    A
CGIA     *         *LBUF*. WRITTEN TO *COMPS* IF C OPTION SELECTED.                    CGIA      2141    A
CGIA     *                                                                             CGIA      2142    A
CGIA     *         PRESERVES - A0, X0, A5, X5                                          CGIA      2143    A
CGIA                                                                                   CGIA      2144    A
CGIA      FMC      ROUTINE                                                             CGIA      2145    A
CGIA               SA2    =1H                                                          CGIA      2146    A
CGIA               SB5    B1          AN = 1        */ ARG NUMBER                      CGIA      2147    A
CGIA               BX7    X2                                                           CGIA      2148    A
CGIA               SA7    LBUF        LBUF(1) = 10H                                    CGIA      2149    A
CGIA               BX4    X1                                                           CGIA      2150    A
CGIA               LX4    59-21                                                        CGIA      2151    A
CGIA               MI     X4,FMC1     IF ^ LAB      */ 1ST ARG NOT IN LABEL FIELD      CGIA      2152    A
CGIA               AX4    56                                                           CGIA      2153    A
CGIA               SA2    PBUF+1                                                       CGIA      2154    A
CGIA               RJ     CMA         CONVERT MACRO ARG                                CGIA      2155    A
CGIA               BFN    X7          BLANK FILL THE NAME                              CGIA      2156    A
CGIA               SX3    B1                        */ SET LOW BIT IN CASE LABEL       CGIA      2157    A
CGIA               BX7    X3+X7                     */ FIELD WAS ABSENT                CGIA      2158    A
CGIA               LX7    -6                                                           CGIA      2159    A
CGIA               SA7    A7          LBUF(1) = SHIFT( CMA(A1)) , 6 )                  CGIA      2160    A
CGIA               SB5    B1+B1       AN = 2                                           CGIA      2161    A
CGIA                                                                                   CGIA      2162    A
CGIA     *         SETUP REGISTERS FOR THE MAIN LOOP                                   CGIA      2163    A
CGIA                                                                                   CGIA      2164    A
CGIA      FMC1     MX2    36                                                           CGIA      2165    A
CGIA               SX3    1R                                                           CGIA      2166    A
CGIA               BX7    X2*X1       ASR = 6L_MAC-NAME  */ ASSEMBLY REGISTER          CGIA      2167    A
CGIA               SB4    18          UBC = 18      */ UNUSED BIT COUNT                CGIA      2168    A
CGIA               LX3    18                                                           CGIA      2169    A
CGIA               BX7    X3+X7       ASR = ASR + SHIFT( 1R  , 18 )                    CGIA      2170    A
CGIA               MX6    -3                                                           CGIA      2171    A
CGIA               BX1    X6*X1                                                        CGIA      2172    A
CGIA               LX1    42          PTW = SHIFT( MHW , 42 ) */ PARAM TYPE BYTES      CGIA      2173    A
CGIA               EQ     B5,B1,FMC2  IF AN = 1                                        CGIA      2174    A
CGIA               LX1    3           PTW = SHIFT( PTW , 3 )                           CGIA      2175    A
CGIA                                                                                   CGIA      2176    A
CGIA      FMC2     SA2    PBUF+B5     ARGV = PBUF(AN)                                  CGIA      2177    A
CGIA               MX6    -3                                                           CGIA      2178    A
CGIA               BX4    -X6*X1      PTB = PTW & 7                                    CGIA      2179    A
CGIA               LX1    3           PTW = SHIFT( PTW , 3 )                           CGIA      2180    A
CGIA               SB5    B5+B1       AN = AN + 1                                      CGIA      2181    A
CGIA               ZR     X4,FMC4     IF PTB = 0    */ END OF ARBS                     CGIA      2182    A
CGIA               BX6    -X6*X1                                                       CGIA      2183    A
CGIA               ZR     X6,FMC3     IF THIS ARG IS THE LAST ARG                      CGIA      2184    A
CGIA               RJ     CMA         CONVERT ARG                                      CGIA      2185    A
CGIA               SA2    FMCA                                                         CGIA      2186    A
CGIA               SB3    B3+6                                                         CGIA      2187    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA              SB2    B3-60                                                  CGIA        2188     A
CGIA              LX2    -B2                                                    CGIA        2189     A
CGIA              BX6    X2+X6                                                  CGIA        2190     A
CGIA              RJ     ADC         ADD *ARG-VAL,* TO STRING                   CGIA        2191     A
CGIA              EQ     FMC2                                                   CGIA        2192     A
CGIA                                                                            CGIA        2193     A
CGIA      FMC3    RJ     CMA         CONVERT LAST ARG                           CGIA        2194     A
CGIA              RJ     ADC                                                    CGIA        2195     A
CGIA                                                                            CGIA        2196     A
CGIA      FMC4    RJ     TSB         TERMINATE BUFFER , WRITE IT TO COMPS       CGIA        2197     A
CGIA              EQ     FMC                                                    CGIA        2198     A
CGIA                                                                            CGIA        2199     A
CGIA      FMCA    DATA   1R,                                                    CGIA        2200     A
CGIA      FBD     SPACE  3,14                                                   CGIA        2201     A
CGIA     **       FBD - FORMAT BINARY DATA. SETS UP FIRST 40 COLUMNS OF PRINT   CGIA        2202     A
CGIA     *                LINE TO - ADDRESS, DATA IN OCTAL AND RELOCATION.     CGIA        2203     A
CGIA     *                CALLED BY A EQ JUMP FROM *APT*                       CGIA        2204     A
CGIA     *                                                                     CGIA        2205     A
CGIA     *        ENTRY  (B2) = PR  , (B3) = NP  , (X4) = DATA                 CGIA        2206     A
CGIA     *                                                                     CGIA        2207     A
CGIA     *        DESTROYS X5                                                  CGIA        2208     A
CGIA                                                                            CGIA        2209     A
CGIA      FBD     SA1    NLF                                                    CGIA        2210     A
CGIA              NZ     X1,APT      IF NLF " 0    */ *FPC*  CALL TO APT        CGIA        2211     A
CGIA                                                                            CGIA        2212     A
CGIA              ZR     B3,FBD4     IF NP = 0                                  CGIA        2213     A
CGIA              GT     B3,B4,FBD5  IF NP > 2                                  CGIA        2214     A
CGIA                                                                            CGIA        2215     A
CGIA              SA1    LBUF                                                   CGIA        2216     A
CGIA              NZ     X1,FBD0     IF LBUF " 0   */ INST CONVERTED TO BCD     CGIA        2217     A
CGIA              SA1    BCDC                                                   CGIA        2218     A
CGIA              ZR     X1,FBD0     IF BCDC = 0                                CGIA        2219     A
CGIA              RJ     FMI         FORMAT MACHINE INSTRUCTION                 CGIA        2220     A
CGIA                                                                            CGIA        2221     A
CGIA      FBD0    NE     B3,B1,FBD3  IF NP = 2                                  CGIA        2222     A
CGIA                                                                            CGIA        2223     A
CGIA              SA1    =5A00000                                               CGIA        2224     A
CGIA              RJ     CDO         CONVERT DATA TO OCTAL                      CGIA        2225     A
CGIA              SX2    B2                                                     CGIA        2226     A
CGIA              SA7    OBUF+3      BLANKS TO RELOC WORD                       CGIA        2227     A
CGIA              LX2    -1                                                     CGIA        2228     A
CGIA              BX3    -X2                                                    CGIA        2229     A
CGIA              PL     X2,FBD1     IF PR IS EVEN                              CGIA        2230     A
CGIA              LX6    30                                                     CGIA        2231     A
CGIA      FBD1    SA6    OBUF+2+X3   STORE NNNNNBBBBB                           CGIA        2232     A
CGIA              SA7    OBUF+1+X2   STORE BLANK WORD                           CGIA        2233     A
CGIA                                                                            CGIA        2234     A
CGIA     *        FORMAT ADDRESS AND LIST THE LINE                             CGIA        2235     A
CGIA                                                                            CGIA        2236     A
CGIA      FBD2    SA1    BCDC                                                   CGIA        2237     A
CGIA              SX5    B3+                                                    CGIA        2238     A
CGIA              MI     X1,FBD2A    IF C " 0                                   CGIA        2239     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                SA1    AI                                                    CGIA        2240     A
CGIA                MX2    -AD.ORGCL                                             CGIA        2241     A
CGIA                SX5    B3                                                    CGIA        2242     A
CGIA                SA3    =1H                                                   CGIA        2243     A
CGIA                UX7    B2,X1                                                 CGIA        2244     A
CGIA                BX1    -X2*X1                                                CGIA        2245     A
CGIA                SB4    B2+B3                                                 CGIA        2246     A
CGIA                BX6    X3          SET RESULT = BLANKS                       CGIA        2247     A
CGIA                SB4    B4-4                                                  CGIA        2248     A
CGIA     +          NZ     B4,*+1      IF PR+NR " 4 THEN OBUF(1) = BLANKS        CGIA        2249     A
CGIA                CALL   COD=              ELSE ADDR                           CGIA        2250     A
CGIA                LX6    6                                                     CGIA        2251     A
CGIA                SA6    OBUF                                                  CGIA        2252     A
CGIA                SA3    LBUF.WC                                               CGIA        2253     A
CGIA                LISTL  A6,X3+4     LIST THE LINE                             CGIA        2254     A
CGIA                SA1    RERR                                                  CGIA        2255     A
CGIA                ZR     X1,FBD2A    IF RERR = 0   */ NO INSTRUCTION ERROR     CGIA        2256     A
CGIA                SX6    X1+3R000                                              CGIA        2257     A
CGIA                LX6    42                                                    CGIA        2258     A
CGIA                MX7    18                                                    CCG16         46     A
CGIA                BX6    X7*X6                                                 CCG16         47     A
CGIA                SA6    A1                                                    CGIA        2259     A
CGIA                LISTL  FBDA,4      -  **** ABOVE INST IN ERROR - XXX -       CGIA        2260     A
CGIA                SX6    0                                                     CGIA        2261     A
CGIA                SA6    RERR        RERR = 0                                  CGIA        2262     A
CGIA                                                                             CGIA        2263     A
CGIA      FBD2A     MX6    0                                                     CGIA        2264     A
CGIA                SA6    LBUF        LBUF = 0                                  CGIA        2265     A
CGIA                LX5    59-1                                                  CGIA        2266     A
CGIA                SA3    AI                                                    CGIA        2267     A
CGIA                SX6    B1                                                    CGIA        2268     A
CGIA                UX7    B2,X3                                                 CGIA        2269     A
CGIA                SA6    LBUF.WC     LBUF.WC = 1                               CGIA        2270     A
CGIA                PL     X5,APT      IF NP " 2                                 CGIA        2271     A
CGIA                EQ     APT6                                                  CGIA        2272     A
CGIA                                                                             CGIA        2273     A
CGIA      FBDA      DATA   30L  ***** ABOVE INST IN ERROR -  XXX                 CGIA        2274     A
CGIA      RERR      BSSZ   1           XOR OF MASK AND REG TYPES                 CGIA        2275     A
CGIA                                                                             CGIA        2276     A
CGIA     *          PROCESS 2 PARCEL DATUM                                      CGIA        2277     A
CGIA                                                                             CGIA        2278     A
CGIA      FBD3      RJ     CRI                                                   CGIA        2279     A
CGIA                SA1    =10H0000000000                                        CGIA        2280     A
CGIA                SA6    OBUF+3                                                CGIA        2281     A
CGIA                RJ     CDO         CONVERT DATA                              CGIA        2282     A
CGIA                SX2    B2                                                    CGIA        2283     A
CGIA                LX2    -1                                                    CGIA        2284     A
CGIA                BX3    -X2                                                   CGIA        2285     A
CGIA                NE     B2,B1,FBD1  IF PR " 1     */ NOT SPLIT WORD CASE      CGIA        2286     A
CGIA                MX3    30                                                    CGIA        2287     A
CGIA                BX7    X3*X7       5L                                        CGIA        2288     A
CGIA                BX4    -X3*X6      L                                         CGIA        2289     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA              BX6    X3*X6       U                                    CGIA        2290     A
CGIA              LX6    30                                               CGIA        2291     A
CGIA              BX6    X7+X6                                            CGIA        2292     A
CGIA              SA6    OBUF+1                                           CGIA        2293     A
CGIA              BX7    X4+X7                                            CGIA        2294     A
CGIA              LX7    30                                               CGIA        2295     A
CGIA              SA7    A6+B1                                            CGIA        2296     A
CGIA              EQ     FBD2                                             CGIA        2297     A
CGIA                                                                     CGIA        2298     A
CGIA     *        PROCESS 0 PARCEL DATUM                                  CGIA        2299     A
CGIA                                                                     CGIA        2300     A
CGIA      FBD4    SA1    =1H                                              CGIA        2301     A
CGIA              BX7    X1                                               CGIA        2302     A
CGIA              SA7    OBUF+1                                           CGIA        2303     A
CGIA              SA7    A7+B1                                            CGIA        2304     A
CGIA              SA7    A7+B1                                            CGIA        2305     A
CGIA              EQ     FBD2                                             CGIA        2306     A
CGIA                                                                     CGIA        2307     A
CGIA     *        PROCESS FULL WORD OF DATA                               CGIA        2308     A
CGIA                                                                     CGIA        2309     A
CGIA      FBD5    RJ     CFW                                              CGIA        2310     A
CGIA              SA7    OBUF+1                                           CGIA        2311     A
CGIA              SA6    A7+B1                                            CGIA        2312     A
CGIA              RJ     CRI         CONVERT RELOC INFO                   CGIA        2313     A
CGIA              SA6    OBUF+3                                           CGIA        2314     A
CGIA              EQ     FBD2                                             CGIA        2315     A
CGIA      CDO     SPACE  2,8                                              CGIA        2316     A
CGIA     **       CDO - CONVERT DATA TO OCTAL                             CGIA        2317     A
CGIA     *                                                                CGIA        2318     A
CGIA     *        ENTRY  (X1) = FILL WORD ( 10H XXXXX )                   CGIA        2319     A
CGIA     *               (X4) = DATA                                      CGIA        2320     A
CGIA     *                                                                CGIA        2321     A
CGIA     *        EXIT   (X6) FILL WORD WITH BCD CONVERSION               CGIA        2322     A
CGIA     *               (X7) = 10H                                       CGIA        2323     A
CGIA                                                                     CGIA        2324     A
CGIA      CDO1    BX7    -X6*X4                                           CGIA        2325     A
CGIA              AX4    3                                                CGIA        2326     A
CGIA              LX7    B5                                               CGIA        2327     A
CGIA              IX1    X1+X7                                            CGIA        2328     A
CGIA              SB5    B5+6                                             CGIA        2329     A
CGIA              NZ     X4,CDO1                                          CGIA        2330     A
CGIA                                                                     CGIA        2331     A
CGIA              BX6    X1                                               CGIA        2332     A
CGIA              SA1    =1H                                              CGIA        2333     A
CGIA              BX7    X1                                               CGIA        2334     A
CGIA                                                                     CGIA        2335     A
CGIA      CDO     ROUTINE                                                 CGIA        2336     A
CGIA              MX6    -3                                               CGIA        2337     A
CGIA              SB5    B0                                               CGIA        2338     A
CGIA              EQ     CDO1                                             CGIA        2339     A
CGIA      CFW     SPACE  2,14                                             CGIA        2340     A
CGIA     **       CFW - CONVERT FULL WORD                                 CGIA        2341     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA     *                                                                  CGIA     2342     A
CGIA     *         ENTRY  (X4) = DATA                                       CGIA     2343     A
CGIA     *                                                                  CGIA     2344     A
CGIA     *         EXIT   (X6,X7) = LOWER 10, UPPER 10 DIGITS               CGIA     2345     A
CGIA                                                                        CGIA     2346     A
CGIA      CFW      ROUTINE                                                  CGIA     2347     A
CGIA               SA1    =10H0000000000                                    CGIA     2348     A
CGIA               MX6    30                                                CGIA     2349     A
CGIA               BX2    -X6*X4                                            CGIA     2350     A
CGIA               LX4    30                                                CGIA     2351     A
CGIA               BX4    -X6*X4                                            CGIA     2352     A
CGIA               RJ     CDO         CONVERT UPPER                         CGIA     2353     A
CGIA               BX3    X6                                                CGIA     2354     A
CGIA               BX4    X2                                                CGIA     2355     A
CGIA               SA1    =10H0000000000                                    CGIA     2356     A
CGIA               RJ     CDO         CONVERT LOWER                         CGIA     2357     A
CGIA               BX7    X3                                                CGIA     2358     A
CGIA               EQ     CFW                                               CGIA     2359     A
CGIA      CRI      SPACE  2,12                                              CGIA     2360     A
CGIA     **        CRI - CONVERT RELOCATION INFORMATION TO DISPLAY CODE     CGIA     2361     A
CGIA     *                                                                  CGIA     2362     A
CGIA     *         EXIT   (X6) = 10H-RELOC-INDICATOR                        CGIA     2363     A
CGIA                                                                        CGIA     2364     A
CGIA      CRI      ROUTINE                                                  CGIA     2365     A
CGIA               SA1    RBWORD                                            CGIA     2366     A
CGIA               LX1    58-AR.RLP                                         CGIA     2367     A
CGIA               PL     X1,CRI1     IF RL < 2                             CGIA     2368     A
CGIA               SA2    =10H  <EXT>                                       CGIA     2369     A
CGIA               LX1    1                                                 CGIA     2370     A
CGIA               MI     X1,CRI2     IF RL = 3                             CGIA     2371     A
CGIA                                                                        CGIA     2372     A
CGIA               SA3    O.PIDL                                            CGIA     2373     A
CGIA               MX6    -AR.RBL                                           CGIA     2374     A
CGIA               LX1    1+AR.RLP-AR.RBP                                   CGIA     2375     A
CGIA               BX6    -X6*X1                                            CGIA     2376     A
CGIA               IX7    X3+X6                                             CGIA     2377     A
CGIA               SA2    X7                                                CGIA     2378     A
CGIA               MX6    -6                                                CGIA     2379     A
CGIA               BX3    -X6*X2                                            CGIA     2380     A
CGIA               BX6    X6*X2                                             CGIA     2381     A
CGIA               SB4    X3-1                                              CGIA     2382     A
CGIA               SA3    =1H                                               CGIA     2383     A
CGIA               MX2    B4                                                CGIA     2384     A
CGIA               BX3    -X2*X3                                            CGIA     2385     A
CGIA               BX6    X6+X3                                             CGIA     2386     A
CGIA               LX6    -6                                                CGIA     2387     A
CGIA               EQ     CRI                                               CGIA     2388     A
CGIA                                                                        CGIA     2389     A
CGIA      CRI1     SA2    =1H                                               CGIA     2390     A
CGIA               LX1    1                                                 CGIA     2391     A
CGIA               PL     X1,CRI2     IF RL = 0                             CGIA     2392     A
CGIA               SA2    =2A+                                              CGIA     2393     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA      CRI2     BX6    X2                                                    CGIA        2394     A
CGIA               EQ     CRI                                                   CGIA        2395     A
CGIA      FMI      TITLE  FMI - FORMAT MACHINE INSTRUCTION                      CGIA        2396     A
CGIA     **        FMI - FORMAT MACHINE INSTRUCTION ( CONVERT TO DPC )          CGIA        2397     A
CGIA     *                                                                      CGIA        2398     A
CGIA     *         ENTRY  (X4) = BINARY                                         CGIA        2399     A
CGIA     *                (B3) = NP ( 1 OR 2 )                                  CGIA        2400     A
CGIA     *                                                                      CGIA        2401     A
CGIA     *         PRESERVES - A0, X0, A5, X5, X4, B2, B3                       CGIA        2402     A
CGIA                                                                            CGIA        2403     A
CGIA                                                                            CGIA        2404     A
CGIA      FMI      ROUTINE                                                      CGIA        2405     A
CGIA               PX6    B3,X4                                                 CGIA        2406     A
CGIA               BX7    X0                                                    CGIA        2407     A
CGIA               SA6    FMIA                                                  CGIA        2408     A
CGIA               SA7    FMIB        SAVE X0, B3 AND X4                        CGIA        2409     A
CGIA               MX1    -3                                                    CGIA        2410     A
CGIA               BX7    -X1*X4      K = BINARY & 7                            CGIA        2411     A
CGIA               EQ     B3,B1,FMI1  IF NP = 1                                 CGIA        2412     A
CGIA               AX4    15          BINARY = SHIFT(BINARY,-15)                CGIA        2413     A
CGIA               SX7    X4          K = SETX(BINARY)                          CGIA        2414     A
CGIA      FMI1     SA7    K                                                     CGIA        2415     A
CGIA               DUP    2,3                                                   CGIA        2416     A
CGIA               AX4    3           BINARY = BINARY / 8                       CGIA        2417     A
CGIA               BX7    -X1*X4      I,J = BINARY & 7                          CGIA        2418     A
CGIA               SA7    A7-B1                                                 CGIA        2419     A
CGIA                                                                            CGIA        2420     A
CGIA               AX4    3           OPC = BINARY/8                            CGIA        2421     A
CGIA               SB5    B1+B1                                                 CGIA        2422     A
CGIA               SB4    X4-1                                                  CGIA        2423     A
CGIA               GT     B4,B5,FMI2  IF OPC > 3                                CGIA        2424     A
CGIA               EQ     B4,B1,FMI4  IF OPC = 2                                CGIA        2425     A
CGIA                                                                            CGIA        2426     A
CGIA               SA1    FMI.XJ+X7   DW = FMI.XJ(I)                            CGIA        2427     A
CGIA               NZ     B4,FMI5     IF OPC = 3                                CGIA        2428     A
CGIA               ZR     X7,FMI4     IF I = 0      */ RJ INSTRUCTION           CGIA        2429     A
CGIA               SA1    FMI.LCM+X7  DW = FMI.LCM(I)                           CGIA        2430     A
CGIA               EQ     FMI5                                                  CGIA        2431     A
CGIA                                                                            CGIA        2432     A
CGIA      FMI2     SX3    X4-50B                                                CGIA        2433     A
CGIA               MI     X3,FMI4     IF OPC < 50B                              CGIA        2434     A
CGIA                                                                            CGIA        2435     A
CGIA               BX2    -X1*X4                                                CGIA        2436     A
CGIA               SA1    FMI.SET+X2  DW = FMI.SET( OPC&7 )                     CGIA        2437     A
CGIA               AX4    3                                                     CGIA        2438     A
CGIA               SX4    X4-4                                                  CGIA        2439     A
CGIA               SX6    X4-3        TRANSLATE F TO A, B OR X                  CGIA        2440     A
CGIA               NZ     X6,FMI3                                               CGIA        2441     A
CGIA               SX4    1RX                                                   CGIA        2442     A
CGIA      FMI3     LX4    48                                                    CGIA        2443     A
CGIA               BX1    X4+X1       DW = DW ! SHIFT(LETTER,48)                CGIA        2444     A
CGIA               EQ     FMI5                                                  CGIA        2445     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                                                                                    CGIA        2446     A
CGIA      FMI4     SA1    FMI.II-1+X4 DW = FMI.II(OPC-1)                                CGIA        2447     A
CGIA                                                                                    CGIA        2448     A
CGIA     *         SETUP REGISTERS FOR MAIN LOOP                                        CGIA        2449     A
CGIA                                                                                    CGIA        2450     A
CGIA      FMI5     SA2    =1H                                                           CGIA        2451     A
CGIA               SB4    48          UBC = 48                                          CGIA        2452     A
CGIA               BX7    X2                                                            CGIA        2453     A
CGIA               SA7    LBUF        LBUF(1) = 1H  ;  SI = 1                           CGIA        2454     A
CGIA               MX3    12                                                            CGIA        2455     A
CGIA               BX7    X3*X1       W = DW & MASK(12)  */ FIRST 2 CHARS               CGIA        2456     A
CGIA               BX1    -X3*X1                                                        CGIA        2457     A
CGIA               LX1    12+5        DW = SHIFT(DW,12+5)                               CGIA        2458     A
CGIA               MX0    -5                                                            CGIA        2459     A
CGIA                                                                                    CGIA        2460     A
CGIA      FMI6     BX6    -X0*X1      C = DW & 37B                                      CGIA        2461     A
CGIA               LX1    5           DW = SHIFT(DW,5)                                  CGIA        2462     A
CGIA               SB6    X6                                                            CGIA        2463     A
CGIA               SB5    6                                                             CGIA        2464     A
CGIA               GE     B6,B5,FMI7  IF C \ 6      */ NO SPECIAL PROCESSING            CGIA        2465     A
CGIA                                                                                    CGIA        2466     A
CGIA               JP     B6+FMI.JT   JUMP( FMI.JT(C) )                                 CGIA        2467     A
CGIA                                                                                    CGIA        2468     A
CGIA      FMI7     SA2    FMI.JT+B6                                                     CGIA        2469     A
CGIA               SB3    6           BC = 6                                            CGIA        2470     A
CGIA               SX3    X6-9                                                          CGIA        2471     A
CGIA               SX6    X2+1R0                                                        CGIA        2472     A
CGIA               LX6    -6          CH = SHIFT(FMI.JT(C)+1R0,54)                      CGIA        2473     A
CGIA               MI     X3,FMI8     IF C < 9      */ REGISTER NUMBER                  CGIA        2474     A
CGIA                                                                                    CGIA        2475     A
CGIA               MX6    6                                                             CGIA        2476     A
CGIA               LX3    1                                                             CGIA        2477     A
CGIA               SA2    FMI.CH                                                        CGIA        2478     A
CGIA               LX4    B1,X3                                                         CGIA        2479     A
CGIA               IX3    X3+X4                                                         CGIA        2480     A
CGIA               SB5    X3                                                            CGIA        2481     A
CGIA               LX2    B5                                                            CGIA        2482     A
CGIA               BX6    X6*X2       CH = MASK(6) & SHIFT(FMI.CH,6*(C-9))              CGIA        2483     A
CGIA                                                                                    CGIA        2484     A
CGIA     *         ADD CHARACTERS TO THE STRING                                         CGIA        2485     A
CGIA                                                                                    CGIA        2486     A
CGIA      FMI8     LX6    B4                                                            CGIA        2487     A
CGIA               BX7    X6+X7       W = W ! SHIFT(CH,UBC)                             CGIA        2488     A
CGIA               SB4    B4-B3       UBC = UBC - BC                                    CGIA        2489     A
CGIA               NZ     B4,FMI6     IF UBC " 0                                        CGIA        2490     A
CGIA               SA7    A7+B1       SI = SI + 1;  LBUF(SI) = W                        CGIA        2491     A
CGIA               MX7    0           W = 0                                             CGIA        2492     A
CGIA               SB4    60          UBC = 60                                          CGIA        2493     A
CGIA               EQ     FMI6                                                          CGIA        2494     A
CGIA      FMI      SPACE  2,10                                                          CGIA        2495     A
CGIA     *         SPECIAL PROCESSORS                                                   CGIA        2496     A
CGIA                                                                                    CGIA        2497     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA     *         Q - ADD ADDRESS EXPRESSION ( IH-H2+CA )                    CGIA        2498     A
CGIA                                                                          CGIA        2499     A
CGIA      FMI.Q    SA2    IH                                                  CGIA        2500     A
CGIA               ZR     X2,FMI.Q2   IF IH = 0                              CGIA        2501     A
CGIA      FMI.Q1   RJ     CSN         CONVERT THE NAME                       CGIA        2502          I
         -CCG16
CGIA      FMI.Q1   CALL   CSN#        CONVERT NAME                           CCG16         48     A
CGIA               RJ     ADC         AND ADD TO THE STRING                  CGIA        2503     A
CGIA               SA2    IH2                                                CGIA        2504     A
CGIA               ZR     X2,FMI.Q3   IF IH2 = 0                             CGIA        2505     A
CGIA               SX6    1R-                                                CGIA        2506     A
CGIA               LX6    -6                                                 CGIA        2507     A
CGIA               SB3    6                                                  CGIA        2508     A
CGIA               RJ     ADC         ADD A -                                CGIA        2509     A
CGIA               SA2    IH2                                                CGIA        2510     A
CGIA               MX6    0                                                  CGIA        2511     A
CGIA               SA6    A2          IH2 = 0                                CGIA        2512     A
CGIA               EQ     FMI.Q1                                             CGIA        2513     A
CGIA                                                                         CGIA        2514     A
CGIA      FMI.Q2   SA2    IH2+1                                              CGIA        2515     A
CGIA               SX1    0           SIGN=0                                 CGIA        2516     A
CGIA               PL     X2,FMI.Q4   IF CA \ 0                              CGIA        2517     A
CGIA                                                                         CGIA        2518     A
CGIA               AX3    B4,X7                                              CGIA        2519     A
CGIA               NZ     X7,FMI.Q2A  GET LAST CHAR IN STRING                CGIA        2520     A
CGIA               SA3    A7                                                 CGIA        2521     A
CGIA                                                                         CGIA        2522     A
CGIA      FMI.Q2A  MX6    -6                                                 CGIA        2523     A
CGIA               BX1    -X6*X3                                             CGIA        2524     A
CGIA               SX4    X1-1R+                                             CGIA        2525     A
CGIA               SX1    1R-         SIGN = -                               CGIA        2526     A
CGIA               NZ     X4,FMI.Q4   IF LAST CHAR " +                       CGIA        2527     A
CGIA               ZR     X7,FMI.Q2B  IF LAST CHAR IS IN PREVIOUS WORD       CGIA        2528     A
CGIA               AX6    B4                                                 CGIA        2529     A
CGIA               SB4    B4+6        UBC = UBC + 6                          CGIA        2530     A
CGIA               LX6    B4                                                 CGIA        2531     A
CGIA               BX7    -X6*X7      REMOVE LAST CHAR                       CGIA        2532     A
CGIA               EQ     FMI.Q4                                             CGIA        2533     A
CGIA                                                                         CGIA        2534     A
CGIA      FMI.Q2B  BX3    X6*X3                                              CGIA        2535     A
CGIA               BX6    X3+X1       CHANGE LAST CHAR TO A -                CGIA        2536     A
CGIA               SA6    A3                                                 CGIA        2537     A
CGIA               MX1    0           SIGN = 0                               CGIA        2538     A
CGIA               EQ     FMI.Q4                                             CGIA        2539     A
CGIA                                                                         CGIA        2540     A
CGIA      FMI.Q3   SA2    IH2+1                                              CGIA        2541     A
CGIA               SX1    1R+         SIGN = 1R+                             CGIA        2542     A
CGIA               PL     X2,FMI.Q4   IF CA \ 0                              CGIA        2543     A
CGIA               SX1    1R-         SIGN = 1R-                             CGIA        2544     A
CGIA                                                                         CGIA        2545     A
CGIA      FMI.Q4   SA2    IH2+1                                              CGIA        2546     A
CGIA               ZR     X2,FMI.END  IF CA = 0                              CGIA        2547     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA              LX1    -6                                                  CGIA        2548     A
CGIA              MX6    0                                                   CGIA        2549     A
CGIA              SA6    A2          IH2(2) = 0                              CGIA        2550     A
CGIA              BX3    X2                                                  CGIA        2551     A
CGIA              AX3    59                                                  CGIA        2552     A
CGIA              BX6    X2-X3                                               CGIA        2553     A
CGIA              RJ     CON         CONVERT CA TO BCD                       CGIA        2554     A
CGIA              BX6    X1+X6                                               CGIA        2555     A
CGIA              NZ     X1,FMI.Q5   IF SIGN " 0                             CGIA        2556     A
CGIA              LX6    6                                                   CGIA        2557     A
CGIA              SB3    B3-6        BC = BC - 6                             CGIA        2558     A
CGIA      FMI.Q5  RJ     ADC         ADD CA TO THE STRING                    CGIA        2559     A
CGIA                                                                         CGIA        2560     A
CGIA      FMI.END RJ     TSB         TERMINATE STRING                        CGIA        2561     A
CGIA              SA4    FMIA                                                CGIA        2562     A
CGIA              SA3    FMIB                                                CGIA        2563     A
CGIA              SA2    AI                                                  CGIA        2564     A
CGIA              UX4    B3,X4                                               CGIA        2565     A
CGIA              BX0    X3                                                  CGIA        2566     A
CGIA              UX6    B2,X2       RESTORE B2                              CGIA        2567     A
CGIA              EQ     FMI                                                 CGIA        2568     A
CGIA                                                                         CGIA        2569     A
CGIA     *        CONDITIONAL B - SKIP AFTER                                CGIA        2570     A
CGIA                                                                         CGIA        2571     A
CGIA      FMI.C   BX6    -X0*X1                                              CGIA        2572     A
CGIA              SA2    FMI.JT+X6                                           CGIA        2573     A
CGIA              LX1    10          DW = SHIFT(DW,10)  */ SKIP NEXT 2 CHARS CGIA        2574     A
CGIA              ZR     X2,FMI6     IF REG NUM OF NEXT = 0                  CGIA        2575     A
CGIA              LX1    -10                                                 CGIA        2576     A
CGIA              R=     X6,1RB                                              CGIA        2577     A
CGIA              SB3    6           BC = 6                                  CGIA        2578     A
CGIA              LX6    54                                                  CGIA        2579     A
CGIA              EQ     FMI8                                                CGIA        2580     A
CGIA                                                                         CGIA        2581     A
CGIA     *        CONDITIONAL B - ADD A +B TO STRING                        CGIA        2582     A
CGIA                                                                         CGIA        2583     A
CGIA      FMI.D   BX3    -X0*X1                                              CGIA        2584     A
CGIA              SX6    2R+B                                                CGIA        2585     A
CGIA              LX6    48          CH = *+B*                               CGIA        2586     A
CGIA              SA2    FMI.JT+X3                                           CGIA        2587     A
CGIA              SB3    12          BC = 12                                 CGIA        2588     A
CGIA              ZR     X2,FMI.END  IF REG-NUM OF NEXT = 0                  CGIA        2589     A
CGIA              RJ     ADC         ADD CHARS                               CGIA        2590     A
CGIA              EQ     FMI6                                                CGIA        2591     A
CGIA                                                                         CGIA        2592     A
CGIA     *        BLANK STRING TO COLUMN 18                                 CGIA        2593     A
CGIA                                                                         CGIA        2594     A
CGIA      FMI.BL  SA2    =1H                                                 CGIA        2595     A
CGIA              SB3    B4-18       BC = UBC - 18                           CGIA        2596     A
CGIA              SB5    B3-B1                                               CGIA        2597     A
CGIA              MX3    B5                                                  CGIA        2598     A
CGIA              BX6    X3*X2       CH = MASK(BC) & 10H                     CGIA        2599     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                EQ     FMI8                                                    CGIA        2600     A
CGIA                                                                               CGIA        2601     A
CGIA     *          SHIFT COUNT - CONVERT TO DECIMAL                                CGIA        2602     A
CGIA                                                                               CGIA        2603     A
CGIA      FMI.S     SA2    J                                                        CGIA        2604     A
CGIA                SA3    A2+B1                                                    CGIA        2605     A
CGIA                LX2    3                                                        CGIA        2606     A
CGIA                BX4    X2+X3                                                    CGIA        2607     A
CGIA                SX6    X4+1R0                                                   CGIA        2608     A
CGIA                LX6    54                                                       CGIA        2609     A
CGIA                SB3    6           BC = 6                                       CGIA        2610     A
CGIA                SX2    X4-10                                                    CGIA        2611     A
CGIA                MI     X2,FMI8     IF SC < 9                                    CGIA        2612     A
CGIA                                                                               CGIA        2613     A
CGIA                SX6    2R10                                                     CGIA        2614     A
CGIA     +          SX2    X2-10                                                    CGIA        2615     A
CGIA                SX6    X6+1S6                                                   CGIA        2616     A
CGIA                PL     X2,*-1                                                   CGIA        2617     A
CGIA                SX6    X6-1S6                                                   CGIA        2618     A
CGIA                SX2    X2+10                                                    CGIA        2619     A
CGIA                IX6    X6+X2                                                    CGIA        2620     A
CGIA                LX6    48                                                       CGIA        2621     A
CGIA                SB3    12          BC = 12                                      CGIA        2622     A
CGIA                EQ     FMI8                                                     CGIA        2623     A
CGIA                                                                               CGIA        2624     A
CGIA     *          COMBINED SPECIAL PROCESSOR JUMP AND CHARACTER DATA TABLE         CGIA        2625     A
CGIA                                                                               CGIA        2626     A
CGIA      FMI.JT    EQ     FMI.END     0                                            CGIA        2627     A
CGIA                EQ     FMI.Q       1                                            CGIA        2628     A
CGIA                EQ     FMI.C       2                                            CGIA        2629     A
CGIA                EQ     FMI.D       3                                            CGIA        2630     A
CGIA                EQ     FMI.BL      4                                            CGIA        2631     A
CGIA                EQ     FMI.S       5                                            CGIA        2632     A
CGIA      I         BSS    1           6                                            CGIA        2633     A
CGIA      J         BSS    1           7                                            CGIA        2634     A
CGIA      K         BSS    1           8                                            CGIA        2635     A
CGIA                                                                               CGIA        2636     A
CGIA      FMI.CH    DATA   8L+-*/,ABX                                               CGIA        2637     A
CGIA      FMI       SPACE  2,10                                                     CGIA        2638     A
CGIA     **         INST - MACRO TO SPECIFY MACHINE INSTRUCTION FORMATS FOR *FMI*   CGIA        2639     A
CGIA     *                                                                         CGIA        2640     A
CGIA     *          INST   (STRING)                                                 CGIA        2641     A
CGIA     *                                                                         CGIA        2642     A
CGIA     *          THE FIRST 2 CHARACTERS HAVE NO SPECIAL MEANING.                 CGIA        2643     A
CGIA     *          CHARACTERS THAT MAY APPEAR IN THE REST OF THE STRING AND        CGIA        2644     A
CGIA     *          THEIR INTERPRETIONS ARE -                                       CGIA        2645     A
CGIA     *          A B X - REGISTER TYPES                                          CGIA        2646     A
CGIA     *          + - * / ,  - SEPERATOR CHARS                                    CGIA        2647     A
CGIA     *          I J K - REGISTER NUMBERS FROM THE MACHINE INSTRUCTION           CGIA        2648     A
CGIA     *          BLANK - END OF OPERATION FIELD                                  CGIA        2649     A
CGIA     *          S - 2 DIGIT SHIFT COUNT ( JK FIELD )                            CGIA        2650     A
CGIA     *          C - CONDITIONAL B-REGISTER SPEC, SKIP IF REG NUM OF NEXT IS =0  CGIA        2651     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA     *                         ELSE ADD A *B* TO THE STRING AND CONTINUE          CGIA     2652     A
CGIA     *         D - CONDITIONAL *+B* SPEC, SKIP IF REG NUM OF NEXT IS =0,            CGIA     2653     A
CGIA     *                         ELSE ADD A *+B* TO THE STRING                      CGIA     2654     A
CGIA     *         Q - ADDRESS EXPRESSION ( IH-IH2+CA , CA = IH2(2) )                   CGIA     2655     A
CGIA     *               APPEARENCE OF A *Q* TERMINATES CONVERSION.                     CGIA     2656     A
CGIA                                                                                    CGIA     2657     A
CGIA              NOREF  .1,.2                                                          CGIA     2658     A
CGIA      INST    MACRO  W                                                              CGIA     2659     A
CGIA      Z       MICRO  1,2,$W$                                                        CGIA     2660     A
CGIA              VFD    12/2R"Z"                                                       CGIA     2661     A
CGIA      .1      SET    3                                                              CGIA     2662     A
CGIA      Z       MICRO  1,,$W$                                                         CGIA     2663     A
CGIA      .2      MICCNT Z                                                              CGIA     2664     A
CGIA              DUP    .2-2                                                           CGIA     2665     A
CGIA      Z       MICRO  .1,1,$W$                                                       CGIA     2666     A
CGIA     *                     ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789+-*/()$= ,          CGIA     2667     A
CGIA      Z       MICRO 1R"Z",1,#NOBC----FGH-----A-E----P------------IJKL----DM#         CGIA     2668     A
CGIA     *                         INDEX INTO FMI.JT                                   CGIA     2669     A
CGIA              VFD    5/1R"Z"                                                        CGIA     2670     A
CGIA      .1      SET    .1+1                                                           CGIA     2671     A
CGIA              ENDD                                                                  CGIA     2672     A
CGIA              VFD    *P/0                                                           CGIA     2673     A
CGIA              ENDM                                                                  CGIA     2674     A
CGIA                                                                                    CGIA     2675     A
CGIA      FMI.XJ  BSS    0          X-JUMPS                                             CGIA     2676     A
CGIA              ECHO   1,ZZ=("XJUMPS")                                                CGIA     2677     A
CGIA              INST   (ZZ XJ,Q)                                                      CGIA     2678     A
CGIA                                                                                    CGIA     2679     A
CGIA      FMI.LCM EQU    *-4        DRL,DWL INSTRUCTIONS                                CGIA     2680     A
CGIA              INST   (RXJ XK)                                                       CGIA     2681     A
CGIA              INST   (WXJ XK)                                                       CGIA     2682     A
CGIA                                                                                    CGIA     2683     A
CGIA      FMI.SET INST   (S:I AJ+Q)                                                     CGIA     2684     A
CGIA              INST   (S:I CJ+Q)                                                     CGIA     2685     A
CGIA              INST   (S:I XJ+Q)                                                     CGIA     2686     A
CGIA              INST   (S:I XJDK)                                                     CGIA     2687     A
CGIA              INST   (S:I AJDK)                                                     CGIA     2688     A
CGIA              INST   (S:I AJ-BK)                                                    CGIA     2689     A
CGIA              INST   (S:I BJDK)                                                     CGIA     2690     A
CGIA              INST   (S:I BJ-BK)                                                    CGIA     2691     A
CGIA                                                                                    CGIA     2692     A
CGIA      FMI.II  BSS    0          OPCODES 1 - 47B                                     CGIA     2693     A
CGIA              INST   (RJ Q)                                                         CGIA     2694     A
CGIA              INST   (JP BI+Q)                                                      CGIA     2695     A
CGIA      FMIA    BSS    1                                                              CGIA     2696     A
CGIA              INST   (EQ CI,CJ,Q)                                                   CGIA     2697     A
CGIA              INST   (NE CI,CJ,Q)                                                   CGIA     2698     A
CGIA              INST   (LE BJ,CI,Q)                                                   CGIA     2699     A
CGIA              INST   (LT BI,CJ,Q)                                                   CGIA     2700     A
CGIA              INST   (BXI XJ)                                                       CGIA     2701     A
CGIA              INST   (BXI XJ*XK)                                                    CGIA     2702     A
CGIA              INST   (BXI XJ+XK)                                                    CGIA     2703     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA              INST   (BXI XJ-XK)                                          CGIA        2704     A
CGIA              INST   (BXI -XJ)                                            CGIA        2705     A
CGIA              INST   (BXI -XK*XJ)                                         CGIA        2706     A
CGIA              INST   (BXI -XK+XJ)                                         CGIA        2707     A
CGIA              INST   (BXI -XK-XJ)                                         CGIA        2708     A
CGIA              INST   (LXI S)                                              CGIA        2709     A
CGIA              INST   (AXI S)                                              CGIA        2710     A
CGIA              INST   (LXI CJ,XK)                                          CGIA        2711     A
CGIA              INST   (AXI CJ,XK)                                          CGIA        2712     A
CGIA              INST   (NXI CJ,XK)                                          CGIA        2713     A
CGIA              INST   (ZXI CJ,XK)                                          CGIA        2714     A
CGIA              INST   (UXI CJ,XK)                                          CGIA        2715     A
CGIA              INST   (PXI CJ,XK)                                          CGIA        2716     A
CGIA              INST   (FXI XJ+XK)                                          CGIA        2717     A
CGIA              INST   (FXI XJ-XK)                                          CGIA        2718     A
CGIA              INST   (DXI XJ+XK)                                          CGIA        2719     A
CGIA              INST   (DXI XJ-XK)                                          CGIA        2720     A
CGIA              INST   (RXI XJ+XK)                                          CGIA        2721     A
CGIA              INST   (RXI XJ-XK)                                          CGIA        2722     A
CGIA              INST   (IXI XJ+XK)                                          CGIA        2723     A
CGIA              INST   (IXI XJ-XK)                                          CGIA        2724     A
CGIA              INST   (FXI XJ*XK)                                          CGIA        2725     A
CGIA              INST   (RXI XJ*XK)                                          CGIA        2726     A
CGIA              INST   (DXI XJ*XK)                                          CGIA        2727     A
CGIA              INST   (MXI S)                                              CGIA        2728     A
CGIA              INST   (FXI XJ/XK)                                          CGIA        2729     A
CGIA              INST   (RXI XJ/XK)                                          CGIA        2730     A
CGIA      FMIB    BSS    1                                                    CGIA        2731     A
CGIA              INST   (CXI XJ)                                             CGIA        2732     A
CGIA      APT     TITLE  LOADER TABLE OUTPUT ROUTINES                         CGIA        2733     A
CGIA     **       APT - ADD PARCELS TO TEXT TABLE                             CGIA        2734     A
CGIA     *                                                                    CGIA        2735     A
CGIA     *        ENTRY  (B3) = NP , NUMBER OF PARCELS OF DATA ( 1 - 4 )      CGIA        2736     A
CGIA     *               (X4) = DATA , RIGHT ADJUSTED                         CGIA        2737     A
CGIA     *               RBWORD - RELOC INFO IN AR. FORMAT                    CGIA        2738     A
CGIA     *               (IH) = SYMTAB ORDINAL                                CGIA        2739     A
CGIA     *                                                                    CGIA        2740     A
CGIA     *        EXIT   DATA ADDED TO TEXT TBL, RELOCATION NOTED IF NP = 2   CGIA        2741     A
CGIA     *                                                                    CGIA        2742     A
CGIA     *        PRESERVES  A5, X0, X5                                       CGIA        2743     A
CGIA                                                                          CGIA        2744     A
CGIA      APT     ROUTINE                                                     CGIA        2745     A
CGIA              SA3    AI                                                   CGIA        2746     A
CGIA              SA1    A3+B1                     (X1) = CW                  CGIA        2747     A
CGIA              UX7    B2,X3       PR = NPR[AI]  */ PARCELS REMAINING IN CW CGIA        2748     A
CGIA              LE     B3,B2,APT4  IF NP @ PR    */ DATA FITS IN CW         CGIA        2749     A
CGIA              ZR     B2,APT2     IF PR = 0     */ CW IS FULL              CGIA        2750     A
CGIA                                                                          CGIA        2751     A
CGIA              SA2    NOPT-1+B2                                            CGIA        2752     A
CGIA              BX1    X2+X1       CW = CW ! NOPT(PR-1)  */ FILL CW WITH NO OP CGIA        2753     A
CGIA                                                                          CGIA        2754     A
CGIA      APT2    SA2    TI                                                   CGIA        2755     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA            BX6    X1          TD(TI) = CW                                  CGIA        2756     A
CGIA            SX1    B1                                                       CGIA        2757     A
CGIA            SA6    TD+X2                                                    CGIA        2758     A
CGIA            SB2    4           PR = 4                                       CGIA        2759     A
CGIA            IX3    X3+X1       ORGC = ORGC + 1                              CGIA        2760     A
CGIA            SX7    X2+B1       TI = TI + 1                                  CGIA        2761     A
CGIA            MX1    0           CW = 0                                       CGIA        2762     A
CGIA            SA7    A2                                                       CGIA        2763     A
CGIA            SX6    X2-14                                                    CGIA        2764     A
CGIA            NZ     X6,APT4     IF TI " 15    */ TEXT TABLE NOT FULL         CGIA        2765     A
CGIA                                                                            CGIA        2766     A
CGIA     *      WRITE TEXT TABLE TO *LGO* AND RESET                             CGIA        2767     A
CGIA                                                                            CGIA        2768     A
CGIA            SA6    A2+         TI = 0                                       CGIA        2769     A
CGIA            SA1    TH                                                       CGIA        2770     A
CGIA            SX2    15+1                                                     CGIA        2771     A
CGIA            SB7    X2+B1                                                    CGIA        2772     A
CGIA            LX2    36                                                       CGIA        2773     A
CGIA            MX6    -12                                                      CCGA068       21     A
CGIA            LX6    36                                                       CCGA068       22     A
CGIA            BX1    X6*X1                                                    CCGA068       23     A
CGIA            BX6    X1+X2       WC[TH] = 16                                  CGIA        2774     A
CGIA            SA6    A1                                                       CGIA        2775     A
CGIA            PX7    B3,X3                                                    CGIA        2776     A
CGIA            SA7    A3                                                       CGIA        2777     A
CGIA            BX6    X4                                                       CGIA        2778     A
CGIA            SA6    A3+B1                                                    CGIA        2779     A
CGIA            WRLGO  A1,B7       TEXT TABLE TO LGO                            CGIA        2780     A
CGIA            SA3    AI                                                       CGIA        2781     A
CGIA            UX7    B3,X3                                                    CGIA        2782     A
CGIA            SA4    A3+B1                   RESTORE DATA                     CGIA        2783     A
CGIA            MX1    0           CW = 0                                       CGIA        2784     A
CGIA            SB2    4           PR = 4                                       CGIA        2785     A
CGIA            MX6    0                                                        CGIA        2786     A
CGIA            SA6    TR          TR = 0                                       CGIA        2787     A
CGIA            SB4    LT.XTXT-2000B                                            CGIA        2788          I
          -CCGA005
CGIA            MX7    -AD.RBOCL                                                CGIA        2789          I
          -CCGA005
CGIA            BX7    -X7*X3                                                   CGIA        2790          I
          -CCGA005
CGIA            PX6    B4,X7                                                    CGIA        2791          I
          -CCGA005
CGIA            SA6    A6-B1       TH = 12/3700B,12/0,12/RB,24/ORGC             CGIA        2792          I
          -CCGA005
CGIA            SETXTH X3,A6-B1                                                 CCGA005       26     A
CGIA                                                                            CGIA        2793     A
CGIA     *      ADD DATA TO *CW*, NOTE RELOCATION IF NP = 2 .                   CGIA        2794     A
CGIA                                                                            CGIA        2795     A
CGIA      APT4  SB2    B2-B3       PR = PR - NP                                 CGIA        2796     A
CGIA            SX2    -B2                                                      CGIA        2797     A
CGIA            LX2    4                                                        CGIA        2798     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA              SB4    X2+B2                                                  CGIA        2799     A
CGIA              LX6    -B4,X4                                                 CGIA        2800     A
CGIA              BX7    X1+X6       CW = CW ! SHIFT(DATA,15*PC)                CGIA        2801     A
CGIA              SA7    A3+B1                                                  CGIA        2802     A
CGIA              PX6    B2,X3                                                  CGIA        2803     A
CGIA      APT5    SA6    A3          ** WORD PLUGGED FOR OLIST*                 CGIA        2804     A
CGIA              SB4    B1+B1                                                  CGIA        2805     A
CGIA              EQ     FBD         (OL ON ) OR NE B3,B4,APT  ( OL OFF )       CGIA        2806     A
CGIA      APT6    SA1    RBWORD                                                 CGIA        2807     A
CGIA              RJ     ATR         ADD TEXT RELOCATION                        CGIA        2808     A
CGIA              EQ     APT                                                    CGIA        2809     A
CGIA                                                                            CGIA        2810     A
CGIA      NOPT    VFD    45/0        TABLE OF NO OPS                            CGIA        2811     A
CGIA     +        VFD    30/0                                                   CGIA        2812     A
CGIA     +        VFD    15/0                                                   CGIA        2813     A
CGIA                                                                            CGIA        2814     A
CGIA      APTA    SA6    A3                                                     CGIA        2815     A
CGIA              SB4    B1+B1                                                  CGIA        2816     A
CGIA              EQ     FBD         PLUG FOR OBJECT LISTING                    CGIA        2817     A
CGIA     +        SA6    A3          PLUG TO RESTORE NO OBJ LISTING             CGIA        2818     A
CGIA              SB4    B1+B1                                                  CGIA        2819     A
CGIA              NE     B3,B4,APT                                              CGIA        2820     A
CGIA      ATR     SPACE  3,14                                                   CGIA        2821     A
CGIA     **       ATR - ADD TEXT RELOCATION ( STANDARD POSITIONS ONLY )         CGIA        2822     A
CGIA     *                                                                      CGIA        2823     A
CGIA     *        ENTRY  (X1) = RBWORD                                           CGIA        2824     A
CGIA     *               (B2) = PR , PARCELS REMAINING IN WORD ( 0 - 2 )        CGIA        2825     A
CGIA     *               (IH) = SYMTAB ORDINAL                                  CGIA        2826     A
CGIA                                                                            CGIA        2827     A
CGIA      ATR     ROUTINE                                                       CGIA        2828     A
CGIA              ZR     X1,ATR      IF RBWORD = 0                              CGIA        2829     A
CGIA              SX6    0           A = RBWORD                                 CGIA        2830     A
CGIA              SA6    RBWORD      RBWORD = 0                                 CGIA        2831     A
CGIA              BX2    X1                                                     CGIA        2832     A
CGIA              LX2    58-AR.RLP                                              CGIA        2833     A
CGIA              LX3    B1,X2                                                  CGIA        2834     A
CGIA              MI     X2,ATR1     IF RL \ 2     */ COMMON OR EXTERNAL        CGIA        2835     A
CGIA              PL     X3,ATR      IF RL = 0     */ ABS                        CGIA        2836     A
CGIA                                                                            CGIA        2837     A
CGIA     *        ADD PROGRAM RELOCATION BYTE TO *TR* WORD IN TEXT TABLE        CGIA        2838     A
CGIA                                                                            CGIA        2839     A
CGIA              SA2    TI                                                     CGIA        2840     A
CGIA              SX7    B2+56                                                  CGIA        2841     A
CGIA              SX4    B1+B1                                                  CGIA        2842     A
CGIA              LX2    2                                                      CGIA        2843     A
CGIA              IX7    X7-X2                                                  CGIA        2844     A
CGIA              SA3    TR                                                     CGIA        2845     A
CGIA              SB3    X7                                                     CGIA        2846     A
CGIA              LX6    B3,X4                                                  CGIA        2847     A
CGIA              BX7    X6+X3       TR = TR ! SHIFT(2,PR+56-4*TI)              CGIA        2848     A
CGIA              SA7    A3                                                     CGIA        2849     A
CGIA              EQ     ATR                                                    CGIA        2850     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                                                                                    CGIA        2851     A
CGIA      ATR1     MI      X3,ATR3     IF RL = 3     */ EXTERNAL                        CGIA        2852     A
CGIA               MX2     -AR.RBL                                                      CGIA        2853     A
CGIA               LX1     -AR.RBP                                                      CGIA        2854     A
CGIA               BX6     -X2*X1      R = RB[A]                                        CGIA        2855     A
CGIA               SA3     O.CBT                                                        CGIA        2856     A
CGIA               LX6     1                                                            CGIA        2857     A
CGIA               SB3     X3+B1                                                        CGIA        2858     A
CGIA               SA4     AI                                                           CGIA        2859     A
CGIA               SA3     B3+X6       LW = CBT(2*R+1)  */ LOADER BLOCK NUMBER          CGIA        2860     A
CGIA               BX7     X4                                                           CGIA        2861     A
CGIA               LX7     59-AD.LCMP                                                   CGIA        2862     A
CGIA               PL      X7,ATR2     IF ^LCM[AI]   */ NOT AN LCM BLOCK                CGIA        2863     A
CGIA                                                                                    CGIA        2864     A
CGIA     *         ADD LINK BYTE FOR LCM RELOC TO XFILL TABLE                           CGIA        2865     A
CGIA                                                                                    CGIA        2866     A
CGIA               SX6     X6+18S10                                                     CGIA        2867     A
CGIA               LX6     9-1                                                          CGIA        2868     A
CGIA               MX3     -AD.ORGCL                                                    CGIA        2869     A
CGIA               BX7     -X3*X4                                                       CGIA        2870     A
CGIA               LX7     30                                                           CGIA        2871     A
CGIA               LX4     -AD.RBP                                                      CGIA        2872     A
CGIA               BX1     -X2*X4                                                       CGIA        2873     A
CGIA               SX2     B2                                                           CGIA        2874     A
CGIA               BX6     X6+X7                                                        CGIA        2875     A
CGIA               LX2     4                                                            CGIA        2876     A
CGIA               BX1     X1+X6                                                        CGIA        2877     A
CGIA               SX3     B2                                                           CGIA        2878     A
CGIA               IX4     X2-X3                                                        CGIA        2879     A
CGIA               LX4     24                                                           CGIA        2880     A
CGIA               BX1     X4+X1       6/0,24/ORGC,6/15*PR,9/RB,9/RB[AI]                CGIA        2881     A
CGIA               ADDWRD  XFT,X1                                                       CGIA        2882     A
CGIA               EQ      ATR                                                          CGIA        2883     A
CGIA                                                                                    CGIA        2884     A
CGIA     *         ADD BYTE TO *LAF* TABLE A3,X3 = LW, X4 = AI                          CGIA        2885     A
CGIA                                                                                    CGIA        2886     A
CGIA      ATR2     SX7     X3          ID = LOW18[LW]                                   CGIA        2887     A
CGIA               SX1     1S2+B2                                                       CGIA        2888     A
CGIA               LX7     30                                                           CGIA        2889     A
CGIA               MX2     -18                                                          CGIA        2890     A
CGIA               LX1     27                                                           CGIA        2891     A
CGIA               BX2     -X2*X4                                                       CGIA        2892     A
CGIA               LX4     -AD.RBP                                                      CGIA        2893     A
CGIA               BX1     X7+X1                                                        CGIA        2894     A
CGIA               MX3     -AD.RBL                                                      CGIA        2895     A
CGIA               BX1     X1+X2                                                        CGIA        2896     A
CGIA               BX4     -X3*X4                                                       CGIA        2897     A
CGIA               LX4     18                                                           CGIA        2898     A
CGIA               BX1     X4+X1       12/,18/ID,1/1,2/PR,9/RB,18/ORGC                  CGIA        2899     A
CGIA               ADDWRD  LAFT,X1                                                      CGIA        2900     A
CGIA               EQ      ATR                                                          CGIA        2901     A
CGIA                                                                                    CGIA        2902     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA     *         SET LW, AI AND GO ADD A LINK BYTE TO *LAFT*                    CGIA        2903     A
CGIA                                                                              CGIA        2904     A
CGIA      ATR3     SA2    IH                                                      CGIA        2905     A
CGIA               SA4    AI                                                      CGIA        2906     A
CGIA               LX3    B1,X2                                                   CGIA        2907     A
CGIA               IX3    X2+X3                                                   CGIA        2908     A
CGIA               SX3    X3+1S16     LW = 1S16+3*IH                              CGIA        2909     A
CGIA               EQ     ATR2                                                    CGIA        2910     A
CGIA      AXR      SPACE  3,14                                                    CGIA        2911     A
CGIA     **        AXR - ADD EXTENDED RELOCATION INFORMATION                      CGIA        2912     A
CGIA     *                                                                        CGIA        2913     A
CGIA     *         ENTRY  (X6) = LEN OF FIELD                                     CGIA        2914     A
CGIA     *                (B2) = BASE, LOW BIT OF FIELD                           CGIA        2915     A
CGIA     *                RBWORD, IH, IH+1 SET                                    CGIA        2916     A
CGIA                                                                              CGIA        2917     A
CGIA      AXRA     CON    1S59+1S44+1S29                                          CGIA        2918     A
CGIA                                                                              CGIA        2919     A
CGIA     *         CALL *ATR* TO ADD STANDARD RELOCATION INFO                     CGIA        2920     A
CGIA                                                                              CGIA        2921     A
CGIA      AXR0     SX7    B2+2                                                    CGIA        2922     A
CGIA               AX7    4                                                       CGIA        2923     A
CGIA               SB2    X7          PR = (BASE+2)/16                            CGIA        2924     A
CGIA               RJ     ATR         ADD TEXT RELOC                              CGIA        2925     A
CGIA                                                                              CGIA        2926     A
CGIA      AXR      ROUTINE                                                        CGIA        2927     A
CGIA               SA2    AXRA                                                    CGIA        2928     A
CGIA               SA1    RBWORD                                                  CGIA        2929     A
CGIA               LX7    B2,X2                                                   CGIA        2930     A
CGIA               PL     X7,AXR1     IF BASE " 0 ! 15 ! 30                       CGIA        2931     A
CGIA               SX7    X6-18                                                   CGIA        2932     A
CGIA               ZR     X7,AXR0     IF LEN = 18                                 CGIA        2933     A
CGIA               PL     X1,AXR0     IF RL = 1     */ LOCAL RELOCATION           CGIA        2934          I
          -CCG16
CGIA                                                                              CGIA        2935     A
CGIA     *         SETUP RELOC WORD FOR XFILL/XLINK TABLE                         CGIA        2936     A
CGIA                                                                              CGIA        2937     A
CGIA      AXR1     SA2    AI                                                      CGIA        2938     A
CGIA               MX7    0                                                       CGIA        2939     A
CGIA               SA7    A1          RBWORD = 0                                  CGIA        2940     A
CGIA               LX6    18                                                      CGIA        2941     A
CGIA               SX7    B2                                                      CGIA        2942     A
CGIA               LX7    24                                                      CGIA        2943     A
CGIA               BX6    X6+X7                                                   CGIA        2944     A
CGIA               MX3    -AD.ORGCL                                               CGIA        2945     A
CGIA               BX7    -X3*X2                                                  CGIA        2946     A
CGIA               LX7    30                                                      CGIA        2947     A
CGIA               BX6    X6+X7                                                   CGIA        2948     A
CGIA               LX2    -AD.RBP                                                 CGIA        2949     A
CGIA               MX3    -AD.RBL                                                 CGIA        2950     A
CGIA               BX7    -X3*X2                                                  CGIA        2951     A
CGIA               BX6    X6+X7       LFW = 30/ORGC[AI],6/BASE,6/LEN,9/0,9/RB[AI] CGIA        2952     A
CGIA               MX7    AR.RLL                                                  CGIA        2953     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA              LX7    AR.RLL+AR.RLP                                          CCG16        49     A
CGIA              BX7    -X1*X7                                                 CGIA       2954     A
CGIA              ZR     X7,AXR3     IF RL = 3     */ EXTERNAL                  CGIA       2955     A
CGIA                                                                            CGIA       2956     A
CGIA     *        ADD XFILL WORD FOR PROGRAM / COMMON SYMBOL                    CGIA       2957     A
CGIA                                                                            CGIA       2958     A
CGIA              SX2    B1          R = 1                                      CGIA       2959     A
CGIA              LX1    58-AR.RLP                                              CGIA       2960     A
CGIA              PL     X1,AXR2     IF RL = 1     */ PROGRAM RELOC             CGIA       2961     A
CGIA              LX1    2+AR.RLP-AR.RBP                                        CGIA       2962     A
CGIA              SA2    CCB                                                    CGIA       2963     A
CGIA              BX3    -X3*X1                                                 CGIA       2964     A
CGIA              IX2    X2+X3       R = CCB + RB[RBWORD]                       CGIA       2965     A
CGIA                                                                            CGIA       2966     A
CGIA      AXR2    LX2    9                                                      CGIA       2967     A
CGIA              BX1    X2+X6       ADD *RB* OF *IH*                           CGIA       2968     A
CGIA              ADDWRD XFT,X1                                                 CGIA       2969     A
CGIA              EQ     AXR                                                    CGIA       2970     A
CGIA                                                                            CGIA       2971     A
CGIA      AXR3    SA2    IH+1                                                   CGIA       2972     A
CGIA              SA6    AXRB+2                                                 CGIA       2973     A
CGIA              BX7    X2                                                     CGIA       2974     A
CGIA              SA7    A6-B1                                                  CGIA       2975     A
CGIA              WRLGO  A7-B1,3     XLINK TABLE TO LGO                         CGIA       2976     A
CGIA              EQ     AXR                                                    CGIA       2977     A
CGIA                                                                            CGIA       2978     A
CGIA      AXRB    VFD    12/LT.XLNK,12/2,36/0                                   CGIA       2979     A
CGIA              BSS    2                                                      CGIA       2980     A
CGIA      DTT     SPACE  3,14                                                   CGIA       2981     A
CGIA     **       DTT - DUMP CURRENT TEXT TABLE PRIOR TO A SWITCH TO ANOTHER    CGIA       2982     A
CGIA     *                BLOCK, ETC.                                          CGIA       2983     A
CGIA     *                                                                     CGIA       2984     A
CGIA     *        EXIT   (X6) = 45/CW,11/0,4/RELOC BYTE                         CGIA       2985     A
CGIA     *                (X3) = AI                                            CGIA       2986     A
CGIA                                                                            CGIA       2987     A
CGIA      DTT     ROUTINE                                                       CGIA       2988     A
CGIA              SA3    AI                                                     CGIA       2989     A
CGIA              SA2    TI                                                     CGIA       2990     A
CGIA              UX7    B2,X3                                                  CGIA       2991     A
CGIA              MX6    0           (X6) = 0      */ ENSURE EXIT CONDITION     CGIA       2992     A
CGIA              NZ     B2,DTT1     IF PR[AI] " 0                              CGIA       2993     A
CGIA              SA1    A3+B1                                                  CGIA       2994     A
CGIA              BX7    X1                                                     CGIA       2995     A
CGIA              SA7    TD+X2       TD(TI) = CW                                CGIA       2996     A
CGIA              SX2    X2+B1       TI = TI + 1                                CGIA       2997     A
CGIA              SA6    A1          CW = 0                                     CGIA       2998     A
CGIA              SB2    4           PR = 4                                     CGIA       2999     A
CGIA              SX4    B1                                                     CGIA       3000     A
CGIA              IX3    X3+X4       ORGC = ORGC + 1                            CGIA       3001     A
CGIA              PX7    B2,X3                                                  CGIA       3002     A
CGIA              SA7    A3                                                     CGIA       3003     A
CGIA                                                                            CGIA       3004     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA      DTT1     ZR     X2,DTT      IF TI = 0     */ NOTHING TO DUMP                CGIA        3005          I
          -CCGA068
CGIA      DTT1     ZR     X2,DTT2     IF TI = 0  NOTHING TO DUMP                      CCGA068       24     A
CGIA               SA1    TH                                                          CGIA        3006     A
CGIA               MX6    -12                                                         CCGA068       25     A
CGIA               LX6    36                                                          CCGA068       26     A
CGIA               BX1    X6*X1                                                       CCGA068       27     A
CGIA               SX2    X2+B1                                                       CGIA        3007     A
CGIA               SB7    X2+B1                                                       CGIA        3008     A
CGIA               LX2    36                                                          CGIA        3009     A
CGIA               BX6    X2+X1       WC[TH] = TI + 1                                 CGIA        3010     A
CGIA               SA6    A1                                                          CGIA        3011     A
CGIA               WRLGO  A1,B7                                                       CGIA        3012     A
CGIA               SA3    AI                                                          CGIA        3013          I
          -CCGA068
CGIA                                                                                 CCGA068       28     A
CGIA      DTT2     SA3    AI                                                          CCGA068       29     A
CGIA               SA1    A3+B1                                                       CGIA        3014     A
CGIA               UX7    B2,X3                                                       CGIA        3015     A
CGIA               SB3    4                                                           CGIA        3016     A
CGIA               SA2    TI                                                          CGIA        3017     A
CGIA               BX6    X1          RETVAL = CW                                     CGIA        3018     A
CGIA               EQ     B2,B3,DTT3  IF PR = 4                                       CGIA        3019     A
CGIA               LX2    2                                                           CGIA        3020     A
CGIA               SA1    TR                                                          CGIA        3021     A
CGIA               SB2    X2+4                                                        CGIA        3022     A
CGIA               MX4    -4                                                          CGIA        3023     A
CGIA               LX1    B2                                                          CGIA        3024     A
CGIA               BX2    -X4*X1                                                      CGIA        3025     A
CGIA               BX6    X2+X6                                                       CGIA        3026     A
CGIA                                                                                 CGIA        3027     A
CGIA      DTT3     SX7    0                                                           CGIA        3028     A
CGIA               SA7    TR          TR = 0                                          CGIA        3029     A
CGIA               SA7    TI          TI = 0                                          CGIA        3030     A
CGIA               EQ     DTT                                                         CGIA        3031     A
CGIA      FPU      SPACE  3,14                                                        CGIA        3032     A
CGIA     **        FPU - FORCE POS COUNTER UPPER                                      CGIA        3033     A
CGIA     *                                                                           CGIA        3034     A
CGIA     *         EXIT   PR = 4 ( AT BEGINNING OF A NEW WORD )                       CGIA        3035     A
CGIA     *                (X3) = ORGC                                                 CGIA        3036     A
CGIA                                                                                 CGIA        3037     A
CGIA      FPU      ROUTINE                                                            CGIA        3038     A
CGIA               SA3    AI                                                          CGIA        3039     A
CGIA               SB4    4                                                           CGIA        3040     A
CGIA               UX7    B3,X3                                                       CGIA        3041     A
CGIA               EQ     B3,B4,FPU   IF PR[AI] = 4                                   CGIA        3042     A
CGIA               SX6    B1                                                          CGIA        3043     A
CGIA               ZR     B3,FPU1     IF PR = 0     */ CURRENT WORD IS FULL           CGIA        3044     A
CGIA                                                                                 CGIA        3045     A
CGIA               SA4    NOPT-1+B3                                                   CGIA        3046     A
CGIA               SA6    NLF         NLF = 1       */ SET NO LIST FLAG               CGIA        3047     A
CGIA               WRTEXT B3          FILL WORD WITH NO OPS                           CGIA        3048     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                MX7    0                                                    CGIA        3049     A
CGIA                SA7    NLF         NLF = 0                                  CGIA        3050     A
CGIA                SX6    B1                                                   CGIA        3051     A
CGIA                SA3    AI                                                   CGIA        3052     A
CGIA                SB4    4                                                    CGIA        3053     A
CGIA                                                                            CGIA        3054     A
CGIA      FPU       IX3    X3+X6       ORGC = ORGC + 1                          CGIA        3055     A
CGIA                PX6    B4,X3       PR = 4                                   CGIA        3056     A
CGIA                SA6    AI                                                   CGIA        3057     A
CGIA                SA2    CW                                                   CGIA        3058     A
CGIA                SA3    TI                                                   CGIA        3059     A
CGIA                MX7    0                                                    CGIA        3060     A
CGIA                BX6    X2                                                   CGIA        3061     A
CGIA                SA6    TD+X3       TD(TI) = CW                              CGIA        3062     A
CGIA                SA7    A2          CW = 0                                   CGIA        3063     A
CGIA                SX6    X3+B1       TI = TI + 1                              CGIA        3064     A
CGIA                SX3    X3-14                                                CGIA        3065     A
CGIA                SA6    A3                                                   CGIA        3066     A
CGIA                NZ     X3,FPU      IF TI " 15    */ TABLE NOT FULL          CGIA        3067     A
CGIA                RJ     DTT         DUMP IT                                  CGIA        3068     A
CGIA                SB4    LT.XTXT-2000B                                        CGIA        3069          I
          -CCGA005
CGIA                MX7    -AD.RBOCL                                            CGIA        3070          I
          -CCGA005
CGIA                BX7    -X7*X3                                               CGIA        3071          I
          -CCGA005
CGIA                PX6    B4,X7                                                CGIA        3072          I
          -CCGA005
CGIA                SA6    TH          TH = 12/LT.XTXT,12/0,36/RBOC[AI]         CGIA        3073          I
          -CCGA005
CGIA                SETXTH X3                                                   CCGA005       27     A
CGIA                EQ     FPU                                                  CGIA        3074     A
CGIA      DAT#      EJECT                                                       CGIA        3075     A
CGIA      **        DAT# - DUMP ACCUMULATED TABLES                              CGIA        3076     A
CGIA      *                DUMPS XFILL, FILL, LINK, ETC TABLES.                CGIA        3077     A
CGIA      *                                                                    CGIA        3078     A
CGIA      *         EXIT   (B5) = SPACE RELEASED                                CGIA        3079     A
CGIA      *                                                                    CGIA        3080     A
CGIA      *         PRESERVES A0,X0, A5,A5                                      CGIA        3081     A
CGIA                                                                            CGIA        3082     A
CGIA      DAT       ENTRY. **,#                                                 CGIA        3083     A
CGIA                SA1    HO$B                                                 CGIA        3084     A
CGIA                ZR     X1,DAT10    IF B = 0                                 CGIA        3085     A
CGIA                                                                            CGIA        3086     A
CGIA      *         DUMP XFILL TABLE                                            CGIA        3087     A
CGIA                                                                            CGIA        3088     A
CGIA                SA4    L.XFT                                                CGIA        3089     A
CGIA                ZR     X4,DAT1     IF L.XFT = 0                             CGIA        3090     A
CGIA                SX7    LT.XFIL                                              CGIA        3091     A
CGIA                LX7    12                                                   CGIA        3092     A
CGIA                BX6    X7+X4                                                CGIA        3093     A
CGIA                LX6    36                                                   CGIA        3094     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA              SA6    T                                                  CGIA        3095     A
CGIA              WRLGO  A6,1                                               CGIA        3096     A
CGIA              SA4    L.XFT                                              CGIA        3097     A
CGIA              SA3    O.XFT                                              CGIA        3098     A
CGIA              WRLGO  X3,X4                                              CGIA        3099     A
CGIA                                                                      CGIA        3100     A
CGIA     *        SORT LINK AND FILL REFS                                  CGIA        3101     A
CGIA                                                                      CGIA        3102     A
CGIA      DAT1    SA1    L.LAFT                                             CGIA        3103     A
CGIA              SA2    O.LAFT                                             CGIA        3104     A
CGIA              SA3    IL.LAF                                             CGIA        3105     A
CGIA              IX1    X1-X3                                              CGIA        3106     A
CGIA              IX2    X2+X3                                              CGIA        3107     A
CGIA              ZR     X1,DAT10     IF L.LAFT = IL.LAF                    CGIA        3108     A
CGIA              IX6    X1+X2                                              CGIA        3109     A
CGIA              MX7    60                                                 CGIA        3110     A
CGIA              SA7    X6           LAFT(L.LAFT)+1) = -0  */ SEARCH TERMINATOR  CGIA  3111     A
CGIA              SB7    X2                                                 CGIA        3112     A
CGIA              CALL   SHL#         SORT( LAFT )                          CGIA        3113     A
CGIA              SA3    O.LAFT                                             CGIA        3114     A
CGIA              SB5    30           SC = 30                               CGIA        3115     A
CGIA              SA2    B7           FI = O.LAFT + IL.LAF                  CGIA        3116     A
CGIA              SB6    X3           WF = O.LAFT                           CGIA        3117     A
CGIA              SB7    B5                                                 CGIA        3118     A
CGIA              MX6    0            W = 0                                 CGIA        3119     A
CGIA              SA6    X3           WI = WF                               CGIA        3120     A
CGIA              MX7    30                                                 CGIA        3121     A
CGIA                                                                      CGIA        3122     A
CGIA     *        FORM THE FILL TABLE                                      CGIA        3123     A
CGIA                                                                      CGIA        3124     A
CGIA      DAT2    LX2    29-16                                              CGIA        3125     A
CGIA              MI     X2,DAT4      IF EXT REF                            CGIA        3126     A
CGIA              LX2    16-29                                              CGIA        3127     A
CGIA              BX4    X2                                                 CGIA        3128     A
CGIA              RJ     AFW          ADD THE WORD                          CGIA        3129     A
CGIA                                                                      CGIA        3130     A
CGIA     *        LOOP TO ADD FILL BYTES                                   CGIA        3131     A
CGIA                                                                      CGIA        3132     A
CGIA      DAT3    BX1    X2           FL = [FI]                             CGIA        3133     A
CGIA              SA2    A2+B1        FI = FI + 1                           CGIA        3134     A
CGIA              BX3    X2-X1                                              CGIA        3135     A
CGIA              BX4    X7*X3                                              CGIA        3136     A
CGIA              NZ     X4,DAT2      IF LBN[FI] " LBN[FL]                  CGIA        3137     A
CGIA              BX3    -X7*X2                                             CGIA        3138     A
CGIA              LX3    B5                                                 CGIA        3139     A
CGIA              BX6    X3+X6        ADD THE BYTE                          CGIA        3140     A
CGIA              SB5    B5-B7                                              CGIA        3141     A
CGIA              PL     B5,DAT3                                            CGIA        3142     A
CGIA              SA6    A6+B1                                              CGIA        3143     A
CGIA              SB5    B7                                                 CGIA        3144     A
CGIA              MX6    0                                                  CGIA        3145     A
CGIA              EQ     DAT3                                               CGIA        3146     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                                                                    CGIA      3147    A
CGIA      DAT4     SX4    LT.FILL                                       CGIA      3148    A
CGIA               RJ     DLT          DUMP THE FILL TABLE              CGIA      3149    A
CGIA                                                                    CGIA      3150    A
CGIA     *         FORM THE LINK TABLE                                  CGIA      3151    A
CGIA                                                                    CGIA      3152    A
CGIA               SA3    O.SYM                                         CGIA      3153    A
CGIA               SB3    X3-1S16                                       CGIA      3154    A
CGIA               SA6    B6           SETUP A6 = WF                    CGIA      3155    A
CGIA      DAT6     ZR     X2,DAT9      IF END OF *LAF*                  CGIA      3156    A
CGIA               AX3    B7,X2                                         CGIA      3157    A
CGIA               SA4    B3+X3                                         CGIA      3158    A
CGIA               MX3    42                                            CGIA      3159    A
CGIA               BX4    X3*X4                                         CGIA      3160    A
CGIA               RJ     AFW          ADD NAME TO LINK TABLE           CGIA      3161    A
CGIA                                                                    CGIA      3162    A
CGIA      DAT7     BX4    -X7*X2                                        CGIA      3163    A
CGIA               LX4    B5                                            CGIA      3164    A
CGIA               BX6    X4+X6                                         CGIA      3165    A
CGIA               SB5    B5-B7                                         CGIA      3166    A
CGIA               PL     B5,DAT8      IF SC \ 0                        CGIA      3167    A
CGIA               SA6    A6+B1                                         CGIA      3168    A
CGIA               SB5    B7           SC = 30                          CGIA      3169    A
CGIA               MX6    0            W = 0                            CGIA      3170    A
CGIA                                                                    CGIA      3171    A
CGIA      DAT8     BX1    X2                                            CGIA      3172    A
CGIA               SA2    A2+B1        FI = FI P 1                      CGIA      3173    A
CGIA               BX3    X1-X2                                         CGIA      3174    A
CGIA               BX4    X7*X3                                         CGIA      3175    A
CGIA               ZR     X4,DAT7      IF EXTN[FI] = EXTN[FL]           CGIA      3176    A
CGIA               EQ     DAT6                                          CGIA      3177    A
CGIA                                                                    CGIA      3178    A
CGIA      DAT9     SX4    LT.LINK                                       CGIA      3179    A
CGIA               RJ     DLT          DUMP THE LINK TABLE              CGIA      3180    A
CGIA                                                                    CGIA      3181    A
CGIA      DAT10    SA4    IL.LAF                                        CGIA      3182    A
CGIA               SA3    L.XFT                                         CGIA      3183    A
CGIA               SA2    L.LAFT                                        CGIA      3184    A
CGIA               SB5    X3                                            CGIA      3185    A
CGIA               MX7    0                                             CGIA      3186    A
CGIA               BX6    X4                                            CGIA      3187    A
CGIA               IX4    X2-X4                                         CGIA      3188    A
CGIA               SB5    B5+X4        SR = L.XFT + L.LAFT-IL.LAF       CGIA      3189    A
CGIA               SA7    A3           L.XFT = 0                        CGIA      3190    A
CGIA               SA6    A2           L.LAFT = IL.LAF                  CGIA      3191    A
CGIA               EQ     DAT                                           CGIA      3192    A
CGIA                                                                    CGIA      3193    A
CGIA      T        OBSS   1            A TEMPORARY                      CGIA      3194    A
CGIA      AFW      SPACE  2,14                                          CGIA      3195    A
CGIA     **        AFW - ADD A FULL WORD TO THE LINK/FILL TABLE         CGIA      3196    A
CGIA     *                                                              CGIA      3197    A
CGIA     *         ENTRY  (X4) = WORD TO BE ADDED                       CGIA      3198    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                                                                    CGIA        3199     A
CGIA      AFW      ROUTINE                                              CGIA        3200     A
CGIA               NE     B5,B7,AFW1                                    CGIA        3201     A
CGIA               BX6    X4                                            CGIA        3202     A
CGIA               SA6    A6+B1                                         CGIA        3203     A
CGIA               MX6    0                                             CGIA        3204     A
CGIA               EQ     AFW                                           CGIA        3205     A
CGIA                                                                    CGIA        3206     A
CGIA      AFW1     BX3    X7*X4                                         CGIA        3207     A
CGIA               LX3    30                                            CGIA        3208     A
CGIA               BX6    X3+X6                                         CGIA        3209     A
CGIA               SA6    A6+B1                                         CGIA        3210     A
CGIA               LX4    30                                            CGIA        3211     A
CGIA               BX6    X7*X4                                         CGIA        3212     A
CGIA               EQ     AFW                                           CGIA        3213     A
CGIA      DLT      SPACE  3,14                                          CGIA        3214     A
CGIA     **        DLT - DUMP LINK/FILL TABLE                           CGIA        3215     A
CGIA     *                                                              CGIA        3216     A
CGIA     *         ENTRY  (X4) = TABLE CODE                             CGIA        3217     A
CGIA                                                                    CGIA        3218     A
CGIA      DLT      ROUTINE                                              CGIA        3219     A
CGIA               EQ     B5,B7,DLT1                                    CGIA        3220     A
CGIA               SA6    A6+1                                          CGIA        3221     A
CGIA      DLT1     SB7    A6+B1                                         CGIA        3222     A
CGIA               SX0    A2                                            CGIA        3223     A
CGIA               SB7    B7-B6                                         CGIA        3224     A
CGIA               SX6    B7-B1                                         CGIA        3225     A
CGIA               ZR     X6,DLT2     IF WC = 1                         CGIA        3226     A
CGIA               LX4    12                                            CGIA        3227     A
CGIA               BX6    X4+X6                                         CGIA        3228     A
CGIA               LX6    36                                            CGIA        3229     A
CGIA               SA6    B6                                            CGIA        3230     A
CGIA               WRLGO  B6,B7                                         CGIA        3231     A
CGIA                                                                    CGIA        3232     A
CGIA      DLT2     SB7    30                                            CGIA        3233     A
CGIA               SB5    B7                                            CGIA        3234     A
CGIA               MX7    30                                            CGIA        3235     A
CGIA               SA2    X0                                            CGIA        3236     A
CGIA               SA3    O.LAFT                                        CGIA        3237     A
CGIA               SB6    X3                                            CGIA        3238     A
CGIA               EQ     DLT                                           CGIA        3239     A
CGIA      GAI      TITLE  UTILITY SUBROUTINES                           CGIA        3240     A
CGIA     **        GAI - GET ADDRESS INFORMATION WORD OF A SYMTAB ENTRY CGIA        3241     A
CGIA     *                                                              CGIA        3242     A
CGIA     *         ENTRY  (X3) = IH , SYMTAB ORDINAL                    CGIA        3243     A
CGIA     *                                                              CGIA        3244     A
CGIA     *         EXIT   (X1) = RBWORD ( WORD C OF SYMTAB )            CGIA        3245     A
CGIA     *                (X3) = RA[RBWORD]                             CGIA        3246     A
CGIA     *                RBWORD, IH, IH+1 AND WB SET                   CGIA        3247     A
CGIA     *                                                              CGIA        3248     A
CGIA     *         USES   B2, X1, X2, X3, X6                            CGIA        3249     A
CGIA                                                                    CCG16         50     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA      GAI0     BX6    -X1*X3                                          CGIA        3250     A
CGIA               AX2    IH.HL                                           CGIA        3251     A
CGIA               SA3    O.GLT-1+X2  BASE = GLT(I[IH])                   CGIA        3252          I
          -CCG16
CGIA               SA2    =XF$FRT-1+X      BASEP = FRT(I[IH])             CCG16         51     A
CGIA               SA3    X2          BASE = [BASEP]                      CCG16         52     A
CGIA               SB2    X6                                              CGIA        3253     A
CGIA               SA1    X3+B2       RBWORD = BASE(H[IH])                CGIA        3254     A
CGIA               BX6    X1                                              CGIA        3255     A
CGIA               SA6    RBWORD                                          CGIA        3256     A
CGIA               MX2    -AR.RAL                                         CGIA        3257     A
CGIA               BX3    -X2*X1                                          CGIA        3258     A
CGIA                                                                      CGIA        3259     A
CGIA      GAI      ROUTINE                                                 CGIA        3260     A
CGIA               MX1    -IH.IHL                                         CGIA        3261     A
CGIA               BX6    -X1*X3                                          CGIA        3262     A
CGIA               MX1    -IH.HL                                          CGIA        3263     A
CGIA               BX2    X1*X6                                           CGIA        3264     A
CGIA               SA6    IH                                              CGIA        3265     A
CGIA               NZ     X2,GAI0     IF I[IH] " 0                        CGIA        3266     A
CGIA                                                                      CGIA        3267     A
CGIA               SA2    O.SYM                                           CGIA        3268     A
CGIA               SB2    X2+2                                            CGIA        3269     A
CGIA               LX3    2                                               CGIA        3270     A
CGIA               IX3    X3-X6                                           CGIA        3271     A
CGIA               SA1    B2+X3       RBWORD = SYM(3*IH+2)                CGIA        3272     A
CGIA      .FPAS    IFNE   HC.FPAS,0                                       CGIA        3273     A
CGIA               SA2    A1-B1                                           CGIA        3274     A
CGIA               BX6    X2          WB = SYM(3*IH+1)                    CGIA        3275     A
CGIA               SA6    WB                                              CGIA        3276     A
CGIA      .FPAS    ENDIF                                                  CGIA        3277     A
CGIA               MX3    AR.RLL                                          CGIA        3278     A
CGIA               LX3    AR.RLL+AR.RLP                                   CCG16         53     A
CGIA               BX3    -X1*X3      RA = 0                              CGIA        3279     A
CGIA               BX6    X1                                              CGIA        3280     A
CGIA               SA6    RBWORD                                          CGIA        3281     A
CGIA               ZR     X3,GAI1     IF RL = 3     */ EXTERNAL           CGIA        3282     A
CGIA               MX3    -AR.RAL                                         CGIA        3283     A
CGIA               BX3    -X3*X1      RA = RA[RBWORD]                     CGIA        3284     A
CGIA               EQ     GAI                                             CGIA        3285     A
CGIA                                                                      CGIA        3286     A
CGIA     *         SAVE NAME IN CASE OF XLINK TABLE ENTRY IS NEEDED       CGIA        3287     A
CGIA                                                                      CGIA        3288     A
CGIA      GAI1     SA2    A1-2                                            CGIA        3289     A
CGIA               MX6    42                                              CGIA        3290     A
CGIA               BX6    X6*X2                                           CGIA        3291     A
CGIA               SA6    IH+1                                            CGIA        3292     A
CGIA               EQ     GAI                                             CGIA        3293     A
CGIA      END      TITLE  PSEUDO INSTRUCTION PROCESSING                   CGIA        3294     A
CGIA     *         PROCESS END OF SUBPROGRAM                              CGIA        3295     A
CGIA     *         END    CA,IH  , CA = LABEL ORDINAL, IH = XFER ORDINAL  CGIA        3296     A
CGIA                                                                      CGIA        3297     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                PROCESS END                                                     CGIA        3298     A
CGIA                SA1    NAE                                                      CGIA        3299     A
CGIA                ZR     X1,END2     IF NAE = 0                                   CGIA        3300     A
CGIA                CALL   CDD=                                                     CGIA        3301     A
CGIA                SA6    NAE                                                      CGIA        3302     A
CGIA                MESSAGE  A6,,R     MESSAGE(  NNN ASSEMBLY ERROR IN PROGNAM )    CGIA        3303     A
CGIA                SA2    =XN$FERR                                                 CGIA        3304     A
CGIA                SX6    X2+B1       N.FERR = N.FERR + 1                          CGIA        3305     A
CGIA                SA6    A2                                                       CGIA        3306     A
CGIA                                                                                CGIA        3307     A
CGIA      END2      RJ     FPU         FORCE UPPER                                  CGIA        3308     A
CGIA                SX6    X5                                                       CGIA        3309     A
CGIA                SA6    ENDB+1      ENDB(2) = IH[II]  */ SAVE XFER NAME ORDINAL  CGIA        3310     A
CGIA                SA2    BCDC                                                     CGIA        3311          I
          -CCG18
CGIA                SA1    N$FERR                                                   CCG18          4     A
CGIA                ZR     X1,END2A    IF N$FERR = 0 */ NO ERRORS                   CCG18          5     A
CGIA                SA2    =XCP.ERCT                                                CCG18          6     A
CGIA                PL     X2,IA.EX    IF CP.ERCT \ 0  */ NOT LGO REGARDLESS        CCG18          7     A
CGIA                                                                                CCG18          8     A
CGIA      END2A     SA2    BCDC                                                     CCG18          9     A
CGIA                ZR     X2,END3     IF BCDC = 0                                  CGIA        3312     A
CGIA                                                                                CGIA        3313     A
CGIA                SA6    PBUF+2      PBUF(2) = IH[II]                             CGIA        3314     A
CGIA                LX5    -SI.CAP                                                  CGIA        3315     A
CGIA                SX6    X5                                                       CGIA        3316     A
CGIA                SA6    A6-B1       PBUF(1) = CA[II]  */ LOC SYMBOL              CGIA        3317     A
CGIA                SA1    ENDA                                                     CGIA        3318     A
CGIA                RJ     FMC         FORMAT END LINE                              CGIA        3319     A
CGIA                WRTEXT 0           AND LIST IT                                  CGIA        3320     A
CGIA                SA1    N$FERR                                                   CGIA        3321          I
          -CCG18
CGIA                NZ     X1,IA.EX    IF N.FERR " 0                                CGIA        3322          I
          -CCG18
CGIA                SA3    BCDC                                                     CGIA        3323     A
CGIA                PL     X3,END3     IF BCDC \ 0   */ C NOT SELECTED              CGIA        3324     A
CGIA                                                                                CGIA        3325     A
CGIA                WRITER F.CMPS,,R   FLUSH COMPS FILE                             CGIA        3326     A
CGIA                EQ     CG$IA                                                    CGIA        3327     A
CGIA                                                                                CGIA        3328     A
CGIA      END3      SA3    HO$B                                                     CGIA        3329     A
CGIA                ZR     X3,CG$IA    IF B = 0                                     CGIA        3330     A
CGIA                                                                                CGIA        3331     A
CGIA                RJ     DTT         DUMP TEXT TABLE                              CGIA        3332     A
CGIA                                                                                CGIA        3333     A
CGIA                RJ     DAT         DUMP XLINK, LINK AND FILL TABLES             CGIA        3334     A
CGIA                                                                                CGIA        3335     A
CGIA                SA3    ENDB+1                                                   CGIA        3336     A
CGIA                ZR     X3,END5     IF ENDB(2) = 0  */ NO XFER NAME              CGIA        3337     A
CGIA                RJ     GAI                                                      CGIA        3338     A
CGIA                SA1    A1-2                                                     CGIA        3339     A
CGIA                MX0    42                                                       CGIA        3340     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA              BX6    X0*X1       ENDB(2) = MASK(42) & SYM(3*IH)                CGIA     3341    A
CGIA              SA6    ENDB+1                                                    CGIA     3342    A
CGIA              WRLGO  A6-B1,2     XFER TABLE TO LGO                             CGIA     3343    A
CGIA                                                                               CGIA     3344    A
CGIA      END5    WRITER F.LGO,,R    FLUSH LGO FILE                                CGIA     3345    A
CGIA              EQ     CG$IA                                                     CGIA     3346    A
CGIA                                                                               CGIA     3347    A
CGIA      ENDA    PSEUDO END,(S,S),1                                               CGIA     3348    A
CGIA      ENDB    VFD    12/LT.XFER,12/1,36/0                                      CGIA     3349    A
CGIA              BSS    1           XFER NAME                                     CGIA     3350    A
CGIA      BTW     SPACE  3,20                                                      CGIA     3351    A
CGIA      VFDW    BSS    2           CURRENT *BTW* TEXT WORD                       CGIA     3352    A
CGIA      VFDL    =      VFDW+1      ADDRESS OF LAST MACRO INST IN THE BTW/ETW     CGIA     3353    A
CGIA     *                           GROUP                                         CGIA     3354    A
CGIA      BTW     SPACE  2,10                                                      CGIA     3355    A
CGIA     **       BTW - BEGIN TEXT WORD DEFINITION                                 CGIA     3356    A
CGIA                                                                               CGIA     3357    A
CGIA              PROCESS BTW                                                      CGIA     3358    A
CGIA              SA4    X5                                                        CGIA     3359    A
CGIA              BX6    X4          VFDW = [LOW18[II]]                            CGIA     3360    A
CGIA              SA6    VFDW                                                      CGIA     3361    A
CGIA              AX5    R1.CAP                                                    CGIA     3362    A
CGIA              SX6    F$MXB+X5                                                  CGIA     3363    A
CGIA              SA6    A6+B1       VFDL = O.MXB + CA[II]                         CGIA     3364    A
CGIA              RJ     FPU         FORCE UPPER                                   CGIA     3365    A
CGIA                                                                               CGIA     3366    A
CGIA     *        GNMW - RETURN POINT FOR PSEUDO INSTRUCTIONS INSIDE A MACRO       CGIA     3367    A
CGIA     *        CHECK FOR END OF *BTW* RANGE.                                    CGIA     3368    A
CGIA                                                                               CGIA     3369    A
CGIA      GNMW    SA4    VFDL                                                      CGIA     3370    A
CGIA              SX5    A5                                                        CGIA     3371    A
CGIA              IX6    X4-X5                                                     CGIA     3372    A
CGIA              NZ     X6,GNMW1    IF II " VFDL                                  CGIA     3373    A
CGIA              SA4    A4-B1                                                     CGIA     3374    A
CGIA              SA6    A4          VFDL = 0                                      CGIA     3375    A
CGIA              WRTEXT 4           OUTPUT CURRENT VFD WORD                       CGIA     3376    A
CGIA                                                                               CGIA     3377    A
CGIA      GNMW1   SA5    A5+B1       II = II + 1                                   CGIA     3378    A
CGIA              UX6    B2,X5                                                     CGIA     3379    A
CGIA              SA4    OC.AT+B2                                                  CGIA     3380    A
CGIA              SB3    X4                                                        CGIA     3381    A
CGIA              AX4    36                                                        CGIA     3382    A
CGIA              JP     B3          JUMP( OC.AT(OC) )                             CGIA     3383    A
CGIA      ARI     SPACE  2,10                                                      CGIA     3384    A
CGIA     **       ARI - ADD RELOCATABLE INFO TO BTW WORD                           CGIA     3385    A
CGIA                                                                               CGIA     3386    A
CGIA              PROCESS ARI                                                      CGIA     3387    A
CGIA              SA2    X5          IH = LOW18[LOW18[II]]                         CGIA     3388    A
CGIA              SX3    X2                                                        CGIA     3389    A
CGIA              SA1    =XS$CON                                                   CGIA     3390    A
CGIA              BX7    X1-X3                                                     CGIA     3391    A
CGIA              SA7    A5          [II] = S$CON - IH   */  =0 IF IH = CON        CGIA     3392    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA              RJ     GAI         GET ADDR INFO                                  CGIA        3393     A
CGIA              RJ     ACF         ADD RA TO VFDW                                 CGIA        3394     A
CGIA              SX6    -B3                                                        CGIA        3395     A
CGIA              RJ     AXR         ADD EXTENDED RELOCATION                        CGIA        3396     A
CGIA              EQ     GNMW                                                       CGIA        3397     A
CGIA       ACI    SPACE  2,10                                                       CGIA        3398     A
CGIA      **      ACI - ADD CONSTANT INFO TO *BTW* WORD                             CGIA        3399     A
CGIA                                                                                CGIA        3400     A
CGIA              PROCESS ACI                                                       CGIA        3401     A
CGIA              SA3    X5          CON = [LOW18[II]]                              CGIA        3402     A
CGIA              RJ     ACF         ADD CONSTANT VALUE TO VFDW                     CGIA        3403     A
CGIA              EQ     GNMW                                                       CGIA        3404     A
CGIA       AAC    SPACE  2,14                                                       CGIA        3405     A
CGIA      **      AAC - ADD ADDRESS CONSTANT                                        CGIA        3406     A
CGIA                                                                                CGIA        3407     A
CGIA              PROCESS AAC                                                       CGIA        3408     A
CGIA              SA4    A5-B1                                                      CGIA        3409     A
CGIA              SA3    X5          CON = [LOW18[II]]                              CGIA        3410     A
CGIA              NZ     X4,AAC1     IF [II-1] " 0 */  LAST IH .NE. CON.            CGIA        3411     A
CGIA              SA1    O.CUT                                                      CGIA        3412     A
CGIA              SB3    X3                                                         CGIA        3413     A
CGIA              SA3    X1+B3       CON = CUT(CON)                                 CGIA        3414     A
CGIA       AAC1   RJ     ACF         ADD VALUE TO FIELD                             CGIA        3415     A
CGIA              EQ     GNMW                                                       CGIA        3416     A
CGIA      *                                                                         CGIA        3417     A
CGIA       ACF    SPACE  3,14                                                       CGIA        3418     A
CGIA      **      ACF - ADD CONSTANT FIELD TO *VFDW*                                CGIA        3419     A
CGIA      *                                                                         CGIA        3420     A
CGIA      *       ENTRY  (X3) = CONSTANT VALUE                                      CGIA        3421     A
CGIA      *              (X5) = 12/P(OC),18/59-LEN,12/P(BASE-BIT),18/XX             CGIA        3422     A
CGIA      *                                                                         CGIA        3423     A
CGIA      *       EXIT   (B2) = BASE BIT                                            CGIA        3424     A
CGIA      *              (B3) = -LEN                                                CGIA        3425     A
CGIA      *                                                                         CGIA        3426     A
CGIA      *       USES   B - 2, 3, 4  X - 2, 3, 5, 6, 7                             CGIA        3427     A
CGIA                                                                                CGIA        3428     A
CGIA       ACF    ROUTINE                                                           CGIA        3429     A
CGIA              LX5    30                                                         CGIA        3430     A
CGIA              SA2    VFDW                                                       CGIA        3431     A
CGIA              UX6    B2,X5                                                      CGIA        3432     A
CGIA              SB3    X5                        (B3) = 59-LEN                    CGIA        3433     A
CGIA              SB4    B2-60                     (B4) = -(60-BASE)                CGIA        3434     A
CGIA              LX2    -B4                                                        CGIA        3435     A
CGIA              MX7    B3                                                         CGIA        3436     A
CGIA              BX6    -X7*X2      EXTRACT                                        CGIA        3437     A
CGIA              BX2    X7*X2       CLEAR                                          CGIA        3438     A
CGIA              SB3    B3+B1                                                      CGIA        3439     A
CGIA              LX3    B3          SIGN EXTEND CON VALUE                          CGIA        3440     A
CGIA              AX3    B3                                                         CGIA        3441     A
CGIA              LX6    B3                                                         CGIA        3442     A
CGIA              AX6    B3          SIGN EXTEND FIELD                              CGIA        3443     A
CGIA              IX3    X3+X6                                                      CGIA        3444     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA              BX6    -X7*X3      TRUNCATE                                  CGIA        3445     A
CGIA              BX6    X2+X6                                                 CGIA        3446     A
CGIA              LX6    B2                                                    CGIA        3447     A
CGIA              SA6    A2                                                    CGIA        3448     A
CGIA              SB3    B3-60             (B3) = -LEN                         CGIA        3449     A
CGIA              EQ     ACF                                                   CGIA        3450     A
CGIA       ASV    SPACE  3,14                                                  CGIA        3451     A
CGIA     **       ASV - ADD STRING VALUE TO CURRENT TEXT WORD                  CGIA        3452     A
CGIA                                                                           CGIA        3453     A
CGIA              PROCESS ASV                                                  CGIA        3454     A
CGIA              SA1    X5          MICRO VALUE                               CGIA        3455     A
CGIA              AX5    18+6                                                  CGIA        3456     A
CGIA              SA2    A1+B1       BIT COUNT OF MICRO                        CGIA        3457     A
CGIA              SB3    X2                                                    CGIA        3458     A
CGIA              SB4    X5          6*NC-1                                    CGIA        3459     A
CGIA              AX5    18                                                    CGIA        3460     A
CGIA              MI     B4,ASV1     IF NC = 0     */ 0L FORMAT                CGIA        3461     A
CGIA                                                                           CGIA        3462     A
CGIA              BX6    X1                                                    CGIA        3463     A
CGIA              BFN    X1          BLANK FILL MICRO                          CGIA        3464     A
CGIA              MX6    B4                                                    CGIA        3465     A
CGIA              BX1    X6*X1       TRUNCATE TO NC CHARS                      CGIA        3466     A
CGIA                                                                           CGIA        3467     A
CGIA       ASV1   SA2    VFDW                                                  CGIA        3468     A
CGIA              MX0    -6                                                    CGIA        3469     A
CGIA              BX4    -X0*X5                                                CGIA        3470     A
CGIA              SB4    X4+B1       TOP BIT + 1                               CGIA        3471     A
CGIA              LX1    B4                                                    CGIA        3472     A
CGIA              BX6    X1+X2                                                 CGIA        3473     A
CGIA              SA6    A2                                                    CGIA        3474     A
CGIA              EQ     GNMW                                                  CGIA        3475     A
CGIA       DCS    SPACE  2,10                                                  CGIA        3476     A
CGIA     **       DCS - DEFINE CHAR STRING AS A MICRO NAME                     CGIA        3477     A
CGIA     *        DCS    CC,I  AND 1 WORD FOLLOWS IN CC_L FORMAT.              CGIA        3478     A
CGIA                                                                           CGIA        3479     A
CGIA              PROCESS DCS                                                  CGIA        3480     A
CGIA              BX0    X5                                                    CGIA        3481     A
CGIA              GNIW                                                         CGIA        3482     A
CGIA              BX6    X5                                                    CGIA        3483     A
CGIA              SX1    X0-1                                                  CGIA        3484     A
CGIA              LX1    1                                                     CGIA        3485     A
CGIA              SA6    =XF$MIC+X1  MIC(2*I-2) = STRING                       CGIA        3486     A
CGIA              AX0    R1.CAP                                                CGIA        3487     A
CGIA              SX1    X0                                                    CGIA        3488     A
CGIA              LX2    B1,X1                                                 CGIA        3489     A
CGIA              IX6    X1+X2                                                 CGIA        3490     A
CGIA              LX6    1                                                     CGIA        3491     A
CGIA              SA6    A6+B1       MIC(2*I-1) = 6*CC                         CGIA        3492     A
CGIA              EQ     GNIW                                                  CGIA        3493     A
CGIA       MIC    SPACE  3,14                                                  CGIA        3494     A
CGIA     *        MIC - DEFINE MICRO STRING                                    CGIA        3495     A
CGIA                                                                           CGIA        3496     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                PROCESS MIC                                                     CGIA        3497     A
CGIA                SA3     X5                                                      CGIA        3498     A
CGIA                SX2     X3          IH = LOW18[LOW18[II]]                       CGIA        3499     A
CGIA                RJ      CSN         GET BCD NAME AND BIT COUNT                  CGIA        3500          I
          -CCG16
CGIA                CALL    CSN#        GET DPC NAME AND BIT COUNT                  CCG16         54     A
CGIA                AX5     18                                                      CGIA        3501     A
CGIA                MX0     -6                                                      CGIA        3502     A
CGIA                BX4     -X0*X5      SEP CHAR                                    CGIA        3503     A
CGIA                LX4     6                                                       CGIA        3504     A
CGIA                BX6     X6+X4                                                   CGIA        3505     A
CGIA                SB6     6                                                       CGIA        3506     A
CGIA                SB2     B0                                                      CGIA        3507     A
CGIA                                                                                CGIA        3508     A
CGIA      *         SEARCH FOR FINAL DELIMETER                                      CGIA        3509     A
CGIA                                                                                CGIA        3510     A
CGIA       MIC1     SB2     B2+B6                                                   CGIA        3511     A
CGIA                LX7     B2,X6                                                   CGIA        3512     A
CGIA                BX4     X5-X7                                                   CGIA        3513     A
CGIA                BX7     -X0*X4                                                  CGIA        3514     A
CGIA                NZ      X7,MIC1                                                 CGIA        3515     A
CGIA                SB2     B2-B6       6*NCHAR IN STRING                           CGIA        3516     A
CGIA                MX2     B2                                                      CGIA        3517     A
CGIA                LX6     -1                                                      CGIA        3518     A
CGIA                BX6     X2*X6       TRUNCATE STRING                             CGIA        3519     A
CGIA                LX6     1                                                       CGIA        3520     A
CGIA                GE      B2,B3,MIC2  BIT COUNT = MIN( BC , 6*NCHARS )            CGIA        3521     A
CGIA                SB3     B2                                                      CGIA        3522     A
CGIA                                                                                CGIA        3523     A
CGIA       MIC2     AX5     6                                                       CGIA        3524     A
CGIA                BX3     -X0*X5      6*FC-6                                      CGIA        3525     A
CGIA                SB4     X3                                                      CGIA        3526     A
CGIA                SB3     B3-B4       BC = BC - (6*FC-6)                          CGIA        3527     A
CGIA                SB2     B4-B1                                                   CGIA        3528     A
CGIA                MX2     B2                                                      CGIA        3529     A
CGIA                BX6     -X2*X6                                                  CGIA        3530     A
CGIA                LX6     B4                                                      CGIA        3531     A
CGIA                                                                                CGIA        3532     A
CGIA       MIC3     AX5     6                                                       CGIA        3533     A
CGIA                BX4     -X0*X5                                                  CGIA        3534     A
CGIA                ZR      X4,MIC4     IF NC = 0                                   CGIA        3535     A
CGIA                BFN     X6                                                      CGIA        3536     A
CGIA                SB3     X4          BC = 6*NC                                   CGIA        3537     A
CGIA                SB4     B3-B1                                                   CGIA        3538     A
CGIA                MX2     B4                                                      CGIA        3539     A
CGIA                BX6     X2*X6       BLANK FILL AND TRUNCATE                     CGIA        3540     A
CGIA                                                                                CGIA        3541     A
CGIA       MIC4     AX5     6                                                       CGIA        3542     A
CGIA                BX5     -X0*X5                                                  CGIA        3543     A
CGIA                SA6     =XF$MIC+X5  STORE NAME                                  CGIA        3544     A
CGIA                SX7     B3                                                      CGIA        3545     A
CGIA                SA7     A6+B1       STORE BIT COUNT                             CGIA        3546     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                EQ     GNMW                                                  CGIA        3547     A
CGIA       IFT      SPACE  3,14                                                  CGIA        3548     A
CGIA      **        IFT - PROCESS IF.XX INSTRUCTIONS                             CGIA        3549     A
CGIA                                                                             CGIA        3550     A
CGIA                PROCESS IFT                                                  CGIA        3551     A
CGIA                MX1    -15                                                   CGIA        3552     A
CGIA                BX6    -X1*X5                                                CGIA        3553     A
CGIA                SA2    PBUF+X6     OP2                                       CGIA        3554     A
CGIA                AX5    15                                                    CGIA        3555     A
CGIA                BX6    -X1*X5                                                CGIA        3556     A
CGIA                SA3    PBUF+X6     OP1                                       CGIA        3557     A
CGIA                SB2    X2                                                    CGIA        3558     A
CGIA                SB3    X3                                                    CGIA        3559     A
CGIA                AX5    15                                                    CGIA        3560     A
CGIA                BX6    -X1*X5      SKIP COUNT                                CGIA        3561     A
CGIA                SB4    X6                                                    CGIA        3562     A
CGIA                MX1    -3                                                    CGIA        3563     A
CGIA                AX5    15                                                    CGIA        3564     A
CGIA                BX5    -X1*X5      TEST TYPE                                 CGIA        3565     A
CGIA                SB5    X5                                                    CGIA        3566     A
CGIA                JP     B5+IFT.JT   JUMP( IFT.JT(IFTYPE) )                    CGIA        3567     A
CGIA       IFT.JT   BSS    0                                                     CGIA        3568     A
CGIA                ECHO   2,Z=("RELOPS")                                        CGIA        3569     A
CGIA                Z      B3,B2,GNMW                                            CGIA        3570     A
CGIA                EQ     IFT2                                                  CGIA        3571     A
CGIA                                                                             CGIA        3572     A
CGIA       IFT2     SA5    A5+B4       SKIP PAST ELSE/ENDIF                      CGIA        3573     A
CGIA                EQ     GNMW                                                  CGIA        3574     A
CGIA       ELSE     SPACE  2,8                                                   CGIA        3575     A
CGIA      **        ELSE - REVERSE EFFECTS OF IF                                 CGIA        3576     A
CGIA                                                                             CGIA        3577     A
CGIA                PROCESS ELSE                                                 CGIA        3578     A
CGIA                SB4    X5                                                    CGIA        3579     A
CGIA                SA5    A5+B4       SKIP UNTIL ENDIF                          CGIA        3580     A
CGIA                EQ     GNMW                                                  CGIA        3581     A
CGIA       SET      EJECT                                                        CGIA        3582     A
CGIA      **        SET -  LI  SET.  ADDR-EXPR                                   CGIA        3583     A
CGIA                                                                             CGIA        3584     A
CGIA                PROCESS SET                                                  CGIA        3585     A
CGIA                SA1    AI                                                    CGIA        3586     A
CGIA                SA2    UBO                                                   CGIA        3587     A
CGIA                MX3    -AD.RBOCL                                             CGIA        3588     A
CGIA                AX2    15                                                    CGIA        3589     A
CGIA                SX2    X2+B1       RL = 1 + SHIFT(UBO,-15)                   CGIA        3590          I
          -CCG29    -CCGA008
CGIA                SX0    B1+B1                                                 CCGA008        4     A
CGIA                IX2    X0-X2       RL = 2 - SHIFT(UB0,-15)                   CCGA008        5     A
CGIA                SX0    B1+B1                                                 CCG29          4          I
          -CCG37
CGIA                IX2    X0-X2       RL = 2 - SHIFT(UBO,-15)                   CCG29          5          I
          -CCG37
CGIA                BX1    -X3*X1                                                CGIA        3591     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA              LX2    AR.RLP                                                CGIA        3592     A
CGIA              BX6    X2+X1                                                 CGIA        3593     A
CGIA              SA6    =XORG#CTR    ORG#CTR = AR(RL,RBOC[AI])                CGIA        3594     A
CGIA              MX0    -15                                                   CGIA        3595     A
CGIA              BX4    -X0*X5                                                CGIA        3596     A
CGIA              RJ     GSA          GET SET ARG VALUES                       CGIA        3597     A
CGIA              AX5    15                                                    CGIA        3598     A
CGIA              BX4    -X0*X5                                                CGIA        3599     A
CGIA              BX6    X7           (X6) = ARG1                              CGIA        3600     A
CGIA              MX1    -3                                                    CGIA        3601     A
CGIA              AX5    15                                                    CGIA        3602     A
CGIA              RJ     GSA          (X7) = ARG2                              CGIA        3603     A
CGIA              BX4    -X1*X5                                                CGIA        3604     A
CGIA              SB4    X4                                                    CGIA        3605     A
CGIA              SB3    60-24                                                 CGIA        3606     A
CGIA              LX2    B3,X6        SIGN EXTEND RA"S                         CGIA        3607     A
CGIA              AX2    B3                                                    CGIA        3608     A
CGIA              LX3    B3,X7                                                 CGIA        3609     A
CGIA              AX3    B3                                                    CGIA        3610     A
CGIA              MX1    AR.RLL                                                CGIA        3611     A
CGIA              LX1    AR.RLL+AR.RLP                                         CCG16         55     A
CGIA              AX5    3                                                     CGIA        3612     A
CGIA              JP     B4+SET.JT                                             CGIA        3613     A
CGIA                                                                           CGIA        3614     A
CGIA      SET.JT  IX3    X2+X3        +                                        CGIA        3615     A
CGIA              EQ     SET1                                                  CGIA        3616     A
CGIA                                                                           CGIA        3617     A
CGIA              IX3    X2-X3        -                                        CGIA        3618     A
CGIA              BX4    X1*X7                                                 CGIA        3619     A
CGIA              EQ     SET.SUB                                               CGIA        3620     A
CGIA                                                                           CGIA        3621     A
CGIA              IX3    X2*X3        *                                        CGIA        3622     A
CGIA              EQ     SET1                                                  CGIA        3623     A
CGIA                                                                           CGIA        3624     A
CGIA     +        ZR     X3,SET1      IF X/0 THEN RETURN 0                     CGIA        3625     A
CGIA              IX3    X2/X3                                                 CGIA        3626     A
CGIA              EQ     SET1                                                  CGIA        3627     A
CGIA                                                                           CGIA        3628     A
CGIA      SET.SUB ZR     X4,SET1      IF RL[OP2] = 0                           CGIA        3629     A
CGIA     *        RL[OP2] = 0 => RL[OP1] = 0                                   CGIA        3630     A
CGIA              MX6    0            (RL,RB)OP1 = 0                           CGIA        3631     A
CGIA              MX7    0            (RL,RB)OP2 = 0                           CGIA        3632     A
CGIA                                                                           CGIA        3633     A
CGIA     *        STORE RESULT VALUE                                           CGIA        3634     A
CGIA                                                                           CGIA        3635     A
CGIA      SET1    MX4    -AR.RAL                                               CGIA        3636     A
CGIA              BX3    -X4*X3       TRUNCATE RA                              CGIA        3637     A
CGIA              BX6    X6+X7        OR RL,RB"S                               CGIA        3638     A
CGIA              BX7    X4*X6                                                 CGIA        3639     A
CGIA              BX6    X7+X3        RECOMBINE RL,RB, & RA                    CGIA        3640     A
CGIA              BX5    -X0*X5       RESULT ORD                               CGIA        3641     A
CGIA              SX3    X5-1S14                                               CGIA        3642     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA              PRINT  SET,(* RESULT,ORD = *O20,Z6),(X6,X5)                CGIA      3643    A
CGIA              PL     X3,SET3     IF ORD \ 1S14 */ RESULT GOES TO SYMTAB  CGIA      3644    A
CGIA              BX2    X4*X6                                               CGIA      3645    A
CGIA              ZR     X2,SET2     IF RL[RESULT] = 0                       CGIA      3646    A
CGIA              SA4    O.SST                                               CGIA      3647        I
          -CCG16
CGIA              SB4    X5+=XSS#BIAS+1S17 I = ORD + SS.BIAS                 CGIA      3648    A
CGIA              SA6    X4+B4       SST(I) = RESULT                         CGIA      3649        I
          -CCG16
CGIA              SA6    F$SST+B4    SST(I) = RESULT                         CCG16       56    A
CGIA              MX7    1           RESULT = 1S59+I  */ SET FLAG BIT        CGIA      3650    A
CGIA              SX6    B4+I.SS                                             CGIA      3651    A
CGIA              BX6    X7+X6       RESULT = 1S59 + I.SS+I                  CGIA      3652    A
CGIA      SET2    SA6    PBUF+X5     PBUF(ORD) = RESULT                      CGIA      3653    A
CGIA              EQ     GNMW                                                CGIA      3654    A
CGIA                                                                         CGIA      3655    A
CGIA      SET3    SA3    PBUF+X3                                             CGIA      3656    A
CGIA              SA2    O.SYM                                               CGIA      3657    A
CGIA              LX4    B1,X3                                               CGIA      3658    A
CGIA              IX3    X3+X4                                               CGIA      3659    A
CGIA              SB3    X2+2                                                CGIA      3660    A
CGIA              SA6    B3+X3       SYM(3*IH+2) = RESULT                    CGIA      3661    A
CGIA              EQ     GNMW                                                CGIA      3662    A
CGIA      GSA     SPACE  2,14                                                CGIA      3663    A
CGIA     **       GSA - GET SET ARGUMENT VALUES                              CGIA      3664    A
CGIA     *                                                                   CGIA      3665    A
CGIA     *        ENTRY  (X4) = ARG INDEX                                    CGIA      3666    A
CGIA     *                                                                   CGIA      3667    A
CGIA     *        EXIT   (X7) = ARG VALUE                                    CGIA      3668    A
CGIA                                                                         CGIA      3669    A
CGIA      GSA0    SA3    PBUF+X3                                             CGIA      3670    A
CGIA              SA2    O.SYM                                               CGIA      3671    A
CGIA              LX4    B1,X3                                               CGIA      3672    A
CGIA              IX3    X3+X4                                               CGIA      3673    A
CGIA              SB3    X2+2                                                CGIA      3674    A
CGIA              SA4    B3+X3       VAL = SYM(3*IH+2)                       CGIA      3675    A
CGIA              BX7    X4                                                  CGIA      3676    A
CGIA      GSA     ROUTINE                                                    CGIA      3677    A
CGIA              SX3    X4-1S14                                             CGIA      3678    A
CGIA              PL     X3,GSA0     IF ARG \ 1S14  */ ARG VAL IN SYMTAB     CGIA      3679    A
CGIA                                                                         CGIA      3680    A
CGIA              SA3    PBUF+X4     VAL = PBUF(ARG)                         CGIA      3681    A
CGIA              SB3    =XORG#CTR                                           CGIA      3682    A
CGIA              BX7    X3                                                  CGIA      3683    A
CGIA              PL     X3,GSA      IF VAL \ 0                              CGIA      3684    A
CGIA              SB3    A3-B3                                               CGIA      3685    A
CGIA              ZR     B3,GSA                                              CGIA      3686    A
CGIA              SA4    O.SST                                               CGIA      3687        I
          -CCG16
CGIA              SB3    X3-I.SS                                             CGIA      3688    A
CGIA              SA3    X4+B3                                               CGIA      3689        I
          -CCG16

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA              SA3    F$SST+B3                                          CCG16        57     A
CGIA              BX7    X3          VAL = SST(VAL-I.SS)                   CGIA       3690     A
CGIA              EQ     GSA                                               CGIA       3691     A
CGIA      LPO     EJECT                                                    CGIA       3692     A
CGIA     **       LPO - LIST PSEUDO OP WITH MAX OF 2 ARGUMENTS             CGIA       3693     A
CGIA     *                                                                 CGIA       3694     A
CGIA     *        ENTRY  (X1) = *FMC* FORMAT CONVERSION WORD               CGIA       3695     A
CGIA     *               (X5) = PSEUDO OP WORD                             CGIA       3696     A
CGIA     *                                                                 CGIA       3697     A
CGIA     *        EXIT   (X5) PRESERVED                                    CGIA       3698     A
CGIA     *               EXITS TO *GNIW* IF *C* SELECTED                   CGIA       3699     A
CGIA                                                                       CGIA       3700     A
CGIA      LPO     ROUTINE                                                  CGIA       3701     A
CGIA              SA2    MIW                                               CGIA       3702     A
CGIA              SA3    BCDC                                              CGIA       3703     A
CGIA              NZ     X2,LPO1     IF MIW " 0    */ IN A MACRO EXPANSION CGIA       3704     A
CGIA              ZR     X3,LPO      IF BCDC = 0   */ OL & C = 0           CGIA       3705     A
CGIA              MX7    -24                                               CGIA       3706     A
CGIA              BX6    -X7*X5                                            CGIA       3707     A
CGIA              SA6    PBUF+2      PBUF(2) = IHX[II]                     CGIA       3708     A
CGIA              LX5    -SI.CAP                                           CGIA       3709     A
CGIA              SX6    X5                                                CGIA       3710     A
CGIA              SA6    A6-B1       PBUF(1) = CA[II]                      CGIA       3711     A
CGIA              LX5    SI.CAP                                            CGIA       3712     A
CGIA              RJ     FMC         FORMAT PSEUDO OP LINE AND WRITE TO *COMPS* CGIA  3713     A
CGIA                                                                       CGIA       3714     A
CGIA      LPO1    SA3    BCDC                                              CGIA       3715     A
CGIA              PL     X3,LPO      IF BCDC \ 0   */ ^ ( C"0 )            CGIA       3716     A
CGIA              EQ     MPXA        */ RETURN TO MAIN LOOP                CGIA       3717     A
CGIA      SLP     SPACE  3,14                                              CGIA       3718     A
CGIA     **       SLP - SUBSTITUTE LOCAL PARAMS IN PSEUDO OP WORD          CGIA       3719     A
CGIA     *                                                                 CGIA       3720     A
CGIA     *        ENTRY  (X1) = *FMC* FORMAT CONVERSION WORD               CGIA       3721     A
CGIA     *        CALLS  LPO                                               CGIA       3722     A
CGIA                                                                       CGIA       3723     A
CGIA      SLP     ROUTINE                                                  CGIA       3724     A
CGIA              BX6    X5                                                CGIA       3725     A
CGIA              LX6    59-24                                             CGIA       3726     A
CGIA              PL     X6,SLP1     IF LOW BITS NOT A LOCAL PARAM         CGIA       3727     A
CGIA                                                                       CGIA       3728     A
CGIA              SA4    X5                                                CGIA       3729     A
CGIA              MX2    -26                                               CGIA       3730     A
CGIA              BX5    X2*X5                                             CGIA       3731     A
CGIA              BX5    X4+X5                                             CGIA       3732     A
CGIA                                                                       CGIA       3733     A
CGIA      SLP1    LX6    24-25                                             CGIA       3734     A
CGIA              LX5    -SI.CAP                                           CGIA       3735     A
CGIA              PL     X6,SLP2     IF CA FIELD NOT A LOCAL PARAM         CGIA       3736     A
CGIA                                                                       CGIA       3737     A
CGIA              SA4    X5                                                CGIA       3738     A
CGIA              MX2    -18                                               CGIA       3739     A
CGIA              BX5    X2*X5                                             CGIA       3740     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                BX5    X4+X5                                          CGIA        3741     A
CGIA                                                                      CGIA        3742     A
CGIA       SLP2     LX5    SI.CAP                                         CGIA        3743     A
CGIA                RJ     LPO         LIST PSEUDO                        CGIA        3744     A
CGIA                EQ     SLP                                            CGIA        3745     A
CGIA       LAB      SPACE  3,14                                           CGIA        3746     A
CGIA      **        LAB    CA,IH                                          CGIA        3747     A
CGIA                                                                      CGIA        3748     A
CGIA                PROCESS LAB                                           CGIA        3749     A
CGIA                SX6    X5                                             CGIA        3750     A
CGIA                SB2    OC.BSS                                         CGIA        3751     A
CGIA                LX6    R1.CAP                                         CGIA        3752     A
CGIA                LX5    -R1.CAP                                        CGIA        3753     A
CGIA                SX5    X5                                             CGIA        3754     A
CGIA                BX6    X5+X6                                          CGIA        3755     A
CGIA                PX5    B2,X6       REFORMAT AS A *BSS*                CGIA        3756     A
CGIA                                                                      CGIA        3757     A
CGIA      **        BSS    SY,*WC                                         CGIA        3758     A
CGIA                                                                      CGIA        3759     A
CGIA                PROCESS BSS                                           CGIA        3760     A
CGIA                SA1    BSSA                                           CGIA        3761     A
CGIA                RJ     SLP         SUBSTITUTE PARAMS AND LIST         CGIA        3762     A
CGIA                MX7    -AR.RAL                                        CGIA        3763     A
CGIA                BX0    -X7*X5      WC = IHX[II]                       CGIA        3764     A
CGIA                RJ     FPU         FORCE UPPER                        CGIA        3765     A
CGIA                                                                      CGIA        3766     A
CGIA      *         CHECK THAT ADDRESS OF LABEL AGREES WITH THE ORG COUNTER.  CGIA    3767     A
CGIA                                                                      CGIA        3768     A
CGIA                LX5    -SI.CAP                                        CGIA        3769     A
CGIA                SX3    X5                                             CGIA        3770     A
CGIA                ZR     X3,BSS1     IF NO SYM                          CGIA        3771     A
CGIA                RJ     GAI                                            CGIA        3772     A
CGIA                SA4    AI                                             CGIA        3773     A
CGIA                MX6    0                                              CGIA        3774     A
CGIA                MX7    -AD.ORGCL                                      CGIA        3775     A
CGIA                SA6    RBWORD                                         CGIA        3776     A
CGIA                BX6    X4-X1                                          CGIA        3777     A
CGIA                BX6    -X7*X6                                         CGIA        3778     A
CGIA                ZR     X6,BSS1     IF ORGC[AI] = RA[IH]               CGIA        3779     A
CGIA                SA2    NAE                                            CGIA        3780     A
CGIA                SX6    X2+B1       NAE = NAE + 1                      CGIA        3781     A
CGIA                SA6    A2                                             CGIA        3782     A
CGIA                SA1    HO$LO$O                                        CGIA        3783     A
CGIA                ZR     X1,BSS1     IF OL = 0                          CGIA        3784     A
CGIA                LISTL  BSSB,BSSC   *  ERROR - RA OF LABEL " ORG.CTR *  CGIA        3785     A
CGIA                                                                      CGIA        3786     A
CGIA       BSS1     SA1    MIW                                            CGIA        3787     A
CGIA                NZ     X1,BSS2     IF MIW " 0    */ INSIDE A MACRO    CGIA        3788     A
CGIA                WRTEXT 0                                              CGIA        3789     A
CGIA       BSS2     ZR     X0,MPXA     IF WC = 0                          CGIA        3790     A
CGIA                                                                      CGIA        3791     A
CGIA                RJ     DTT         DUMP THE CURRENT TEXT TABLE        CGIA        3792     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                IX6    X3+X0       ORGC[AI] = ORGC[AI] + WC                    CGIA        3793     A
CGIA                SA6    A3                                                      CGIA        3794     A
CGIA                SB3    LT.XTXT-2000B                                           CGIA        3795          I
          -CCGA005
CGIA                MX4    -AD.RBOCL                                               CGIA        3796          I
          -CCGA005
CGIA                BX6    -X4*X6                                                  CGIA        3797          I
          -CCGA005
CGIA                PX7    B3,X6                                                   CGIA        3798          I
          -CCGA005
CGIA                SA7    TH          TH = 12/LT.XTXT,12/,36/RBOC[AI]             CGIA        3799          I
          -CCGA005
CGIA                SETXTH X6                                                      CCGA005       28     A
CGIA                EQ     MPXA                                                    CGIA        3800     A
CGIA                                                                              CGIA        3801     A
CGIA      BSSA      PSEUDO BSS,(S,C),1                                             CGIA        3802     A
CGIA      BSSB      DIS    ,/ ERROR - RA OF LABEL .NE. ORG.CTR/                    CGIA        3803     A
CGIA      BSSC      EQU    *-BSSB                                                  CGIA        3804     A
CGIA      BSSZ      SPACE  3,14                                                    CGIA        3805     A
CGIA     **         BSSZ   SY,*WC                                                  CGIA        3806     A
CGIA                                                                              CGIA        3807     A
CGIA                PROCESS BSSZ                                                   CGIA        3808     A
CGIA                SA1    BSSZA                                                   CGIA        3809     A
CGIA                RJ     SLP         SUBSTITUTE PARAMS AND LIST                  CGIA        3810     A
CGIA                MX7    -AR.RAL                                                 CGIA        3811     A
CGIA                BX0    -X7*X5      WC = IHX[SI]                                CGIA        3812     A
CGIA                RJ     FPU         FORCE UPPER                                 CGIA        3813     A
CGIA                ZR     X0,MPXA     IF WC = 0                                   CGIA        3814     A
CGIA                SA3    AI                                                      CGIA        3815     A
CGIA                MX6    -AD.RBOCL                                               CGIA        3816     A
CGIA                BX7    -X6*X3      SA = RBOC[AI] */ SOURCE ADDR                CGIA        3817     A
CGIA                SA7    BSSZB+1                                                 CGIA        3818     A
CGIA                MX4    0                                                       CGIA        3819     A
CGIA                WRTEXT 4           OUTPUT FIRST ZERO WORD                      CGIA        3820     A
CGIA                SX6    B1                                                      CGIA        3821     A
CGIA                IX0    X0-X6       WC = WC - 1                                 CGIA        3822     A
CGIA                ZR     X0,MPXA     IF WC = 0     */ ONLY 1 WORD                CGIA        3823     A
CGIA                SA6    NLF         NLF = 1       SHUT OFF LISTING              CGIA        3824     A
CGIA                BX7    X0                                                      CGIA        3825     A
CGIA                AX7    2                                                       CGIA        3826     A
CGIA                NZ     X7,BSSZ3    IF WC > 3                                   CGIA        3827     A
CGIA                                                                              CGIA        3828     A
CGIA      BSSZ1     MX4    0                                                       CGIA        3829     A
CGIA                WRTEXT 4           OUTPUT A ZERO WORD                          CGIA        3830     A
CGIA                SX0    X0-1        WC = WC - 1                                 CGIA        3831     A
CGIA                NZ     X0,BSSZ1    IF WC " 0                                   CGIA        3832     A
CGIA                                                                              CGIA        3833     A
CGIA      BSSZ2     SX6    0           NLF = 0                                     CGIA        3834     A
CGIA                SA6    NLF                                                     CGIA        3835     A
CGIA                EQ     MPXA                                                    CGIA        3836     A
CGIA                                                                              CGIA        3837     A
CGIA     *          WC > 4, OUTPUT 1 OR MORE *REP* TABLES                          CGIA        3838     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                                                                                    CGIA        3839     A
CGIA      BSSZ3    RJ     DTT         FLUSH TEXT TABLE                                  CGIA        3840     A
CGIA               IX6    X0+X3       ORG[AI] = ORG[AI] + WC                            CGIA        3841     A
CGIA               SA6    A3                                                            CGIA        3842          I
          -CCGA007
CGIA               SB3    LT.XTXT-2000B                                                 CGIA        3843          I
          -CCGA007 -CCGA005
CGIA               MX4    -AD.RBOCL                                                     CGIA        3844          I
          -CCGA007 -CCGA005
CGIA               BX6    -X4*X6                                                        CGIA        3845          I
          -CCGA007 -CCGA005
CGIA               PX7    B3,X6                                                         CGIA        3846          I
          -CCGA007 -CCGA005
CGIA               MX1    -15                                                           CGIA        3847          I
          -CCGA007 -CCGA005
CGIA               SA7    TH          TH = 12/LT.XTXT,12/0,36/RBOC[AI]                  CGIA        3848          I
          -CCGA007 -CCGA005
CGIA               SETXTH X6                                                            CCGA005       29          I
          -CCGA007
CGIA               MX1    -15                                                           CCGA005       30          I
          -CCGA007
CGIA               BX2    X1*X0                                                         CGIA        3849          I
          -CCGA007
CGIA               SA6    A3                                                            CCGA007        4     A
CGIA               SETXTH X6                                                            CCGA007        5     A
CGIA               SA2    BSSZB                                                         CCGA007        6     A
CGIA               MX4    -AD.RBL                                                       CCGA007        7     A
CGIA               LX4    12                                                            CCGA007        8     A
CGIA               BX6    X4*X2       CLEAR RB NUMBER FROM CL FIELD                     CCGA007        9     A
CGIA               LX3    59-AD.CLP                                                     CCGA007       10     A
CGIA               PL     X3,BSSZ3A                                                     CCGA007       11     A
CGIA               LX3    1+AD.CLP-AD.RBP+12                                            CCGA007       12     A
CGIA               BX3    -X4*X3                                                        CCGA007       13     A
CGIA               BX6    X3+X6       SET RB NUMBER IN CL FIELD                         CCGA007       14     A
CGIA      BSSZ3A   SA6    A2                                                            CCGA007       15     A
CGIA               MX1    -15                                                           CCGA007       16     A
CGIA               BX2    X1*X0                                                         CCGA007       17     A
CGIA               BX6    X0                                                            CGIA        3850     A
CGIA               LX6    45                                                            CGIA        3851     A
CGIA               ZR     X2,BSSZ4    IF WC < 2**15                                     CGIA        3852     A
CGIA                                                                                    CGIA        3853     A
CGIA               MX1    -9                                                            CGIA        3854     A
CGIA               BX4    -X1*X0                                                        CGIA        3855     A
CGIA               SX6    X4+1S9      RC1 = WC & 777B + 1000B                           CGIA        3856     A
CGIA               LX6    45                                                            CGIA        3857     A
CGIA               SA6    BSSZB+2                                                       CGIA        3858     A
CGIA               WRLGO  BSSZB,3     OUTPUT FIRST REP TABLE                            CGIA        3859     A
CGIA               SA1    BSSZB+1                                                       CGIA        3860     A
CGIA               SA2    A1+B1                                                         CGIA        3861     A
CGIA               LX2    15                                                            CGIA        3862     A
CGIA               SX3    X2+B1                                                         CGIA        3863     A
CGIA               IX6    X1+X3       DA = SA + RC1+1                                   CGIA        3864     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA              SX7    1S9          BLOCK SIZE = 1000B                          CGIA     3865    A
CGIA              LX7    33                                                       CGIA     3866    A
CGIA              BX6    X7+X6                                                    CGIA     3867    A
CGIA              IX0    X0-X2                                                    CGIA     3868    A
CGIA              AX0    9            REP COUNT = (WC-RC1)/1000B                  CGIA     3869    A
CGIA              LX0    45                                                       CGIA     3870    A
CGIA              BX6    X6+X0                                                    CGIA     3871    A
CGIA                                                                             CGIA     3872    A
CGIA      BSSZ4   SA6    BSSZB+2                                                  CGIA     3873    A
CGIA              WRLGO  BSSZB,3      OUTPUT REP TABLE                            CGIA     3874    A
CGIA              EQ     BSSZ2                                                    CGIA     3875    A
CGIA                                                                             CGIA     3876    A
CGIA      BSSZA   PSEUDO BSSZ,(S,C),1                                             CGIA     3877    A
CGIA      BSSZB   VFD    12/LT.REPI,12/2,36/1                                     CGIA     3878    A
CGIA              BSS    2                                                        CGIA     3879    A
CGIA      BOS     SPACE  2,14                                                     CGIA     3880    A
CGIA     **       BOS    LINE-NUM,0   , H2 = 2/LIST-FLAGS,10/XX                   CGIA     3881    A
CGIA     *        LISTS A LINE OF THE FORM *35X,"LINE ",IN*                       CGIA     3882    A
CGIA                                                                             CGIA     3883    A
CGIA              PROCESS BOS                                                     CGIA     3884    A
CGIA              SA4    BCDC                                                     CGIA     3885    A
CGIA              LX6    -SI.CAP                                                  CGIA     3886    A
CGIA              SX1    X6           LINEN = CA[II]                              CGIA     3887    A
CGIA              SX7    X6+                                                      CGIA     3888    A
CGIA              SA7    =XHO$CSN     HO$CSN = LINEN                              CGIA     3889    A
CGIA              MI     X4,BOS2      IF C " 0                                    CGIA     3890    A
CGIA              LX5    -28                                                      CGIA     3891    A
CGIA              MX4    -2                                                       CGIA     3892    A
CGIA              BX3    -X4*X5                                                   CGIA     3893    A
CGIA              SB2    X3           LF = H2[II]   */ LIST FLAGS                 CGIA     3894    A
CGIA              ZR     B2,BOS1      IF LF = 0     */ NO CHANGE                  CGIA     3895    A
CGIA              SB3    B1+B1                                                    CGIA     3896    A
CGIA              SX6    B3-B2                                                    CGIA     3897    A
CGIA              SA6    A4           BCDC = 2 - LF                               CGIA     3898    A
CGIA              SA3    APTA-1+B2                                                CGIA     3899    A
CGIA              BX7    X3           ADJUST PLUG                                 CGIA     3900    A
CGIA              SA7    APT5                                                     CGIA     3901    A
CGIA      BOS1    SA4    A4+                                                      CGIA     3902    A
CGIA              ZR     X4,GNIW      IF BCDC = 0                                 CGIA     3903    A
CGIA                                                                             CGIA     3904    A
CGIA      BOS2    MX0    -12                                                      CGIA     3905    A
CGIA              CALL   CDD=         RETURNS (X4) = LEFT JUSTIFIED NUMBER        CGIA     3906    A
CGIA              SA1    =1H                                                      CGIA     3907    A
CGIA              SA2    =5ALINE                                                  CGIA     3908    A
CGIA              BX7    X1                                                       CGIA     3909    A
CGIA              SA7    LBUF         FOR I = 1 TO 3 ; LBUF(I) = 1H               CGIA     3910    A
CGIA              SA7    A7+B1                                                    CGIA     3911    A
CGIA              SA7    A7+B1                                                    CGIA     3912    A
CGIA              BX7    X2                                                       CGIA     3913    A
CGIA              SA7    A7+B1        LBUF(4) = 5ALINE                            CGIA     3914    A
CGIA              BX7    X0*X4        WD = 8LLINE-NUM                             CGIA     3915    A
CGIA              SB4    12           UBC = 12                                    CGIA     3916    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA              RJ     TSB         TERMINATE LINE AND WRITE TO COMPS          CGIA     3917     A
CGIA              WRTEXT 0           LIST THE LINE IF OL " 0                    CGIA     3918     A
CGIA              EQ     MPXA                                                   CGIA     3919     A
CGIA       CON    SPACE  2,14                                                   CGIA     3920     A
CGIA     **       CON    SY,*CA                                                 CGIA     3921     A
CGIA                                                                            CGIA     3922     A
CGIA              PROCESS CON                                                   CGIA     3923     A
CGIA              SX7    1R+                                                    CGIA     3924     A
CGIA              SA7    FMCA        SET SEPERATOR = +                          CGIA     3925     A
CGIA              SA1    CONA                                                   CGIA     3926     A
CGIA              RJ     SLP         SUBSTITUTE LOCAL PARAMS AND FORMAT         CGIA     3927     A
CGIA              SX7    1R,                                                    CGIA     3928     A
CGIA              SA7    FMCA        RESTORE SEPERATOR CHAR                     CGIA     3929     A
CGIA              MX7    -AR.RAL                                                CGIA     3930     A
CGIA              BX4    -X7*X5      DATA = IHX[II]                             CGIA     3931     A
CGIA              LX5    -SI.CAP                                                CGIA     3932     A
CGIA              SX3    X5                                                     CGIA     3933     A
CGIA              RJ     GAI                                                    CGIA     3934     A
CGIA              IX4    X3+X4       DATA = DATA + RA[RBWORD]                   CGIA     3935     A
CGIA              WRTEXT 4                                                      CGIA     3936     A
CGIA              SX6    AR.RAL                                                 CGIA     3937     A
CGIA              SB2    B0                                                     CGIA     3938     A
CGIA              RJ     AXR         ADD RELOCATION                             CGIA     3939     A
CGIA              EQ     GNIW                                                   CGIA     3940     A
CGIA                                                                            CGIA     3941     A
CGIA       CONA   PSEUDO CON,(S,C)                                              CGIA     3942     A
CGIA       DATA   SPACE  2,14                                                   CGIA     3943     A
CGIA     **       DATA   WC,CC AND WC WORDS FOLLOW                             CGIA     3944     A
CGIA                                                                            CGIA     3945     A
CGIA              PROCESS DATA                                                  CGIA     3946     A
CGIA              BX0    X5                                                     CGIA     3947     A
CGIA              AX0    R1.CAP                                                 CGIA     3948     A
CGIA              SX0    X0          WC = CA[II]                                CGIA     3949     A
CGIA              SX5    X5                                                     CGIA     3950     A
CGIA              NZ     X0,DATA1    IF WC " 0                                  CGIA     3951     A
CGIA              SA1    DATAA                                                  CGIA     3952     A
CGIA              LX5    R1.CAP                                                 CGIA     3953     A
CGIA              RJ     LPO         LIST PSEUDO                                CGIA     3954     A
CGIA              LX5    -R1.CAP                                                CGIA     3955     A
CGIA              SX4    X5                                                     CGIA     3956     A
CGIA              WRTEXT 4                                                      CGIA     3957     A
CGIA              EQ     GNIW                                                   CGIA     3958     A
CGIA                                                                            CGIA     3959     A
CGIA       DATA1  GNIW               GET A DATA WORD                            CGIA     3960     A
CGIA              SA1    BCDC                                                   CGIA     3961     A
CGIA              ZR     X1,DATA2    IF BCDC = 0                                CGIA     3962     A
CGIA                                                                            CGIA     3963     A
CGIA              BX4    X5                                                     CGIA     3964     A
CGIA              RJ     CFW         CONVERT 20 DIGITS                          CGIA     3965     A
CGIA              BX1    X6          SAVE LOWER                                 CGIA     3966     A
CGIA              BX6    X7                                                     CGIA     3967     A
CGIA              SA2    =1H                                                    CGIA     3968     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                BX7    X2                                                  CGIA        3969     A
CGIA                SA3    =8LDATA                                             CGIA        3970     A
CGIA                SA7    LBUF        LBUF(1) = 10H                           CGIA        3971     A
CGIA                SB3    60          BC = 60                                 CGIA        3972     A
CGIA                SB4    12          UBC = 12                                CGIA        3973     A
CGIA                BX7    X3          WD = 8LDATA                             CGIA        3974     A
CGIA                RJ     ADC         ADD UPPER TO STRING                     CGIA        3975     A
CGIA                BX6    X1                                                  CGIA        3976     A
CGIA                R=     X1,1RB                                              CGIA        3977     A
CGIA                RJ     ADC         ADD LOWER                               CGIA        3978     A
CGIA                LX1    6                                                   CGIA        3979     A
CGIA                SB4    6           UBC = 6                                 CGIA        3980     A
CGIA                BX7    X1+X7       ADD FINAL B                             CGIA        3981     A
CGIA                RJ     TSB         TERMINATE STRING AND WRITE TO *COMPS*   CGIA        3982     A
CGIA       DATA2    BX4    X5                                                  CGIA        3983     A
CGIA                WRTEXT 4                                                   CGIA        3984     A
CGIA                SX0    X0-1        WC = WC - 1                             CGIA        3985     A
CGIA                NZ     X0,DATA1    IF WC " 0                               CGIA        3986     A
CGIA                EQ     GNIW                                                CGIA        3987     A
CGIA                                                                           CGIA        3988     A
CGIA       DATAA    PSEUDO DATA,C                                              CGIA        3989     A
CGIA       DIS      SPACE  2,14                                                CGIA        3990     A
CGIA      *         DIS    0,WC  , AND WC WORDS FOLLOW IN *H* FORMAT           CGIA        3991     A
CGIA                                                                           CGIA        3992     A
CGIA                PROCESS DIS                                                CGIA        3993     A
CGIA                SX6    X5                                                  CGIA        3994     A
CGIA                SA6    DISA        WC = IH[II]                             CGIA        3995     A
CGIA                                                                           CGIA        3996     A
CGIA       DIS1     SA4    DISA                                                CGIA        3997     A
CGIA                SX5    5                                                   CGIA        3998     A
CGIA                IX7    X4-X5                                               CGIA        3999     A
CGIA                PL     X7,DIS2     WTM = MIN( WC , 5 )  */ WORDS TO MOVE   CGIA        4000     A
CGIA                BX5    X4                                                  CGIA        4001     A
CGIA       DIS2     IX6    X4-X5       WC = WC - WTM                           CGIA        4002     A
CGIA                SA6    A4                                                  CGIA        4003     A
CGIA                ZR     X5,GNIW     IF WTM = 0                              CGIA        4004     A
CGIA                SA1    =1H                                                 CGIA        4005     A
CGIA                SA2    =10HDIS     0,                                      CGIA        4006     A
CGIA                BX6    X1                                                  CGIA        4007     A
CGIA                SA6    LBUF                                                CGIA        4008     A
CGIA                LX5    6                                                   CGIA        4009     A
CGIA                IX7    X5+X2             ADD WC TO DIS STMT                CGIA        4010     A
CGIA                SA7    A6+B1                                               CGIA        4011     A
CGIA                LX5    -6                                                  CGIA        4012     A
CGIA                SA0    A7+B1                                               CGIA        4013     A
CGIA                RJ     MWI         MWI( WTM , LBUF+2 )                     CGIA        4014     A
CGIA                SX7    1R                                                  CGIA        4015     A
CGIA                SB4    60-6                                                CGIA        4016     A
CGIA                LX7    -6                                                  CGIA        4017     A
CGIA                RJ     TSB         TERMINATE THE STRING AND WRITE TO *COMPS*  CGIA     4018     A
CGIA                                                                           CGIA        4019     A
CGIA      *         NOW OUTPUT THE BINARY, NOTE THAT *NLF* SERVES A DOUBLE USE  CGIA       4020     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                                                                                    CGIA        4021     A
CGIA      DIS3     SA1    NLF                                                           CGIA        4022     A
CGIA               SA4    LBUF+2+X1    DATA = LBUF(NLF+2)                               CGIA        4023     A
CGIA               WRTEXT 4            OUTPUT BINARY AND LIST FIRST TIME                CGIA        4024     A
CGIA               SA1    NLF                                                           CGIA        4025     A
CGIA               SX6    X1+B1        NLF = NLF + 1                                    CGIA        4026     A
CGIA               SA6    A1                                                            CGIA        4027     A
CGIA               SX5    A0-LBUF-2                                                     CGIA        4028     A
CGIA               IX7    X6-X5                                                         CGIA        4029     A
CGIA               MI     X7,DIS3      IF NLF < WTM                                     CGIA        4030     A
CGIA               SA7    A1           NLF = 0                                          CGIA        4031     A
CGIA               EQ     DIS1                                                          CGIA        4032     A
CGIA                                                                                    CGIA        4033     A
CGIA      DISA     OBSS   1                                                             CGIA        4034     A
CGIA      ENT      SPACE  2                                                             CGIA        4035     A
CGIA               PROCESS ENT                                                          CGIA        4036     A
CGIA               SX4    B1                                                            CGIA        4037     A
CGIA               LX4    SI.CAP                                                        CGIA        4038     A
CGIA               BX5    X4+X5        CA = 1                                           CGIA        4039     A
CGIA               EQ     .LAB                                                          CGIA        4040     A
CGIA      HOL      EJECT                                                                CGIA        4041     A
CGIA     **        HOL    CC,F    CC = CHAR COUNT, F = 1RH, 1RL OR 1RR                 CGIA        4042     A
CGIA     *                            AND 1 WORD FOLLOWS IN H FORMAT                   CGIA        4043     A
CGIA                                                                                    CGIA        4044     A
CGIA               PROCESS HOL                                                          CGIA        4045     A
CGIA               SA3    =1H                                                           CGIA        4046     A
CGIA               SA1    =9LHOL      0                                                 CGIA        4047     A
CGIA               SX4    X6                                                            CGIA        4048     A
CGIA               AX6    SI.CAP-6                                                      CGIA        4049     A
CGIA               BX7    X3                                                            CGIA        4050     A
CGIA               IX6    X1+X6                                                         CGIA        4051     A
CGIA               SA7    LBUF                                                          CGIA        4052     A
CGIA               BX6    X6+X4        10LHOL      NF                                   CGIA        4053     A
CGIA               SA6    A7+B1                                                         CGIA        4054     A
CGIA               UX0    X5                                                            CGIA        4055     A
CGIA               GNIW                GET DATA WORD                                    CGIA        4056     A
CGIA               BX7    X5                                                            CGIA        4057     A
CGIA               SA7    LBUF+2                                                        CGIA        4058     A
CGIA               SB4    12                                                            CGIA        4059     A
CGIA               MX7    0                                                             CGIA        4060     A
CGIA               MX5    0                                                             CCGA005       31     A
CGIA               RJ     TSB         TERMINATE LINE AND WRITE TO *COMPS*                CGIA        4061     A
CGIA               SB2    X0-1RH                                                        CGIA        4062     A
CGIA               BX4    X5                                                            CGIA        4063     A
CGIA               AX0    SI.CAP-1    2*CC                                              CGIA        4064     A
CGIA               ZR     B2,HOL1     IF F = 1RH                                        CGIA        4065     A
CGIA               LX1    B1,X0                                                         CGIA        4066     A
CGIA               IX2    X1+X0       6*CC                                              CGIA        4067     A
CGIA               SB4    X2                                                            CGIA        4068     A
CGIA               SB3    B4-B1                                                         CGIA        4069     A
CGIA               MX6    B3                                                            CGIA        4070     A
CGIA               BX4    X6*X4       REMOVE TRAILING BLANKS                            CGIA        4071     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                SB2    B2+1RH-1RL                                          CGIA        4072     A
CGIA                ZR     B2,HOL1     IF F = 1RL                              CGIA        4073     A
CGIA                LX4    B4                                                  CGIA        4074     A
CGIA      HOL1      WRTEXT 4           OUTPUT BINARY                           CGIA        4075     A
CGIA                EQ     GNIW                                                CGIA        4076     A
CGIA      ORG       SPACE  3,14                                                CGIA        4077     A
CGIA     **         ORG    IH,CAX   - SET ORG COUNTER                          CGIA        4078     A
CGIA                                                                           CGIA        4079     A
CGIA      .Z        SET    -1                                                  CGIA        4080     A
CGIA                IFC    NE,/"HC.UDVB"//,4                                   CGIA        4081     A
CGIA                ECHO   3,X=("HC.FLB")                                      CGIA        4082     A
CGIA      .Z        SET    .Z+1                                                CGIA        4083     A
CGIA                IFC    EQ,/X/"HC.UDVB"/,1                                  CGIA        4084     A
CGIA                STOPDUP                                                    CGIA        4085     A
CGIA      HC.UDVB   MAX    0,.Z                                                CGIA        4086     A
CGIA                                                                           CGIA        4087     A
CGIA      K.        VFD    2/1,25/0,9/HC.UDVB,24/0   SAVED VALUE OF UDV BLOCK ADDRESS  CGIA  4088     A
CGIA                                                                           CGIA        4089     A
CGIA                PROCESS ORG                                                CGIA        4090     A
CGIA                UX6    X5                                                  CGIA        4091     A
CGIA                NZ     X6,ORG1     IF CAIH[II] " 0                         CGIA        4092     A
CGIA                SA1    K.                                                  CGIA        4093     A
CGIA                RJ     SOC         RESET ORG CTR TO PREVIOUS VALUE         CGIA        4094     A
CGIA                MX7    39          M=MASK(39)                              CGIA        4095     A
CGIA                EQ     ORG2                                                CGIA        4096     A
CGIA                                                                           CGIA        4097     A
CGIA      ORG1      SX6    1S15+HC.UDVB                                        CGIA        4098     A
CGIA                SX5    0                                                   CCGA005       32     A
CGIA                RJ     CUB         CHANGE BLOCKS                           CGIA        4099     A
CGIA                RJ     FPU         FORCE UPPER                             CGIA        4100     A
CGIA                SA1    K.                                                  CGIA        4101     A
CGIA                SA2    AI                                                  CGIA        4102     A
CGIA                MX6    X1+X2       K. = MAX( K. , AI )                     CGIA        4103     A
CGIA                SA6    A1                                                  CGIA        4104     A
CGIA                SA5    A5                                                  CGIA        4105     A
CGIA                MX0    -AR.RAL                                             CGIA        4106     A
CGIA                BX4    -X0*X5                                              CGIA        4107     A
CGIA                LX5    -SI.CAP                                             CGIA        4108     A
CGIA                SX3    X5                                                  CGIA        4109     A
CGIA                RJ     GAI         WC = GAI( IH[II] )                      CGIA        4110     A
CGIA                IX1    X1+X4       RA[WC] = RA[WC] + CAL[II]               CGIA        4111     A
CGIA                RJ     SOC         SET NEW ORG VALUE                       CGIA        4112     A
CGIA                SA6    S.          SAVE FOR REPI PROCESSOR                 CGIA        4113     A
CGIA                MX7    60          M = MASK(60)                            CGIA        4114     A
CGIA                                                                           CGIA        4115     A
CGIA      ORG2      SA2    ORGA                                                CGIA        4116     A
CGIA                SA5    A5                                                  CGIA        4117     A
CGIA                BX1    X7*X2       FMCW = ORGA & M                         CGIA        4118     A
CGIA                RJ     LPO         LIST THE PSEUDO                         CGIA        4119     A
CGIA                WRTEXT 0                                                   CGIA        4120     A
CGIA                EQ     GNIW                                                CGIA        4121     A
CGIA                                                                           CGIA        4122     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA      ORGA     PSEUDO ORG,(S,C)                                          CGIA        4123     A
CGIA      SOC      SPACE  2,14                                               CGIA        4124     A
CGIA     **        SOC - SET ORG COUNTER                                     CGIA        4125     A
CGIA     *                                                                  CGIA        4126     A
CGIA     *         ENTRY  (X1) = NORG, ADDRESS IN AR. FORMAT                 CGIA        4127     A
CGIA     *                                                                  CGIA        4128     A
CGIA     *         EXIT   (X6) = ADDRESS IN AD. FORMAT                       CGIA        4129     A
CGIA                                                                         CGIA        4130     A
CGIA      SOC0     MX0    -AD.RBOCL                                          CGIA        4131     A
CGIA               BX6    -X0*X4                                             CGIA        4132     A
CGIA      SOC      ROUTINE                                                   CGIA        4133     A
CGIA               SA2    UBO                                                CGIA        4134     A
CGIA               LX1    AR.RLL                                             CGIA        4135          I
          -CCG16
CGIA               MX0    -AR.RBL                                            CGIA        4136     A
CGIA               BX3    -X0*X2       RB = UBO & 777B                       CGIA        4137     A
CGIA               AX2    15                                                 CGIA        4138     A
CGIA               SX7    B1+B1                                              CGIA        4139     A
CGIA               IX6    X7-X2        RL = 2 - UBO/1S15                     CGIA        4140     A
CGIA               SA4    AI                                                 CGIA        4141     A
CGIA               MX0    -AD.ORGCL                                          CGIA        4142     A
CGIA               BX5    -X0*X4                                             CGIA        4143     A
CGIA               LX3    AR.RBP       CORG = AR(RL,RB,ORGC[AI])             CGIA        4144     A
CGIA               BX5    X3+X5                                              CGIA        4145     A
CGIA               LX5    AR.RLL                                             CGIA        4146          I
          -CCG16
CGIA               LX6    AR.RLP                                             CCG16         58     A
CGIA               BX6    X5+X6                                              CGIA        4147     A
CGIA               BX7    X6-X1                                              CGIA        4148     A
CGIA               MX0    -AR.RLL-AD.RBOCL                                   CGIA        4149          I
          -CCG16
CGIA               SA2    SOCB                                               CCG16         59     A
CGIA               BX7    X2*X7                                              CCG16         60     A
CGIA               BX7    -X0*X7                                             CGIA        4150          I
          -CCGA068
CGIA               LX1    -AR.RLL                                            CGIA        4151          I
          -CCG16
CGIA               ZR     X7,SOC0      IF CORG = NORG                        CGIA        4152     A
CGIA                                                                         CGIA        4153     A
CGIA               BX6    X1                                                 CGIA        4154     A
CGIA               SA6    SOCA                                               CGIA        4155     A
CGIA               MX7    -AR.RBL                                            CGIA        4156     A
CGIA               LX6    -AR.RBP                                            CGIA        4157     A
CGIA               BX6    -X7*X6       NUBO = RB[NORG]                       CGIA        4158     A
CGIA               LX1    58-AR.RLP                                          CCG16         61     A
CGIA     +         MI     X1,*+1                                             CGIA        4159     A
CGIA               SX6    X6+1S15      IF( RL = 1 ) THEN UBO = UBO + 1S15    CGIA        4160     A
CGIA               RJ     CUB          SWITCH TO THE NEW BLOCK               CGIA        4161     A
CGIA               NZ     X7,*+1       IF NO CHANGE                          CGIA        4162     A
CGIA               RJ     DTT          DUMP THE TEXT TABLE IF NO CHANGE      CGIA        4163     A
CGIA               SA1    SOCA                                               CGIA        4164     A
CGIA               SA2    AI                                                 CGIA        4165     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA              SA3    TH                                                  CGIA        4166     A
CGIA              MX0    -AD.ORGCL                                           CGIA        4167     A
CGIA              BX1    -X0*X1                                              CGIA        4168     A
CGIA              BX2    X0*X2                                               CGIA        4169     A
CGIA              IX6    X1+X2       ORGC[AI] = ORGC[NORG]                   CGIA        4170     A
CGIA              SA6    A2                                                  CGIA        4171     A
CGIA              BX3    X0*X3                                               CGIA        4172     A
CGIA              IX6    X1+X3       ORGC[TH] = ORGC[NORG]                   CGIA        4173     A
CGIA              SA6    A3                                                  CGIA        4174     A
CGIA              MX7    -AD.RBOCL                                           CGIA        4175     A
CGIA              BX6    -X7*X6      RBOC = RBOC[TH]                         CGIA        4176     A
CGIA              EQ     SOC                                                 CGIA        4177     A
CGIA                                                                         CGIA        4178     A
CGIA      SOCA    OBSS   1                                                   CGIA        4179     A
CGIA      SOCB    BFMW   AR,(RL,RB,RA)                                       CCG16         62     A
CGIA      REPI    SPACE  2,14                                                CGIA        4180     A
CGIA     **       REPI   DLEN,RC,INC,DESTIN-ADDR-INC                         CGIA        4181     A
CGIA                                                                         CGIA        4182     A
CGIA              PROCESS REPI                                               CGIA        4183     A
CGIA              SX6    X5                                                  CGIA        4184     A
CGIA              SA6    PBUF+2      REP COUNT                               CGIA        4185     A
CGIA              LX5    -SI.CAP                                             CGIA        4186     A
CGIA              SX7    X5                                                  CGIA        4187     A
CGIA              SA7    A6-B1       DLEN                                    CGIA        4188     A
CGIA              LX6    45                                                  CGIA        4189     A
CGIA              LX7    AD.RBOCL                                            CGIA        4190     A
CGIA              BX0    X6+X7                                               CGIA        4191     A
CGIA              GNIW               GET SECOND WORD                         CGIA        4192     A
CGIA              SX6    X5                                                  CGIA        4193     A
CGIA              SA6    PBUF+3      INC                                     CGIA        4194     A
CGIA              AX5    18                                                  CGIA        4195     A
CGIA              SX7    X5                                                  CGIA        4196     A
CGIA              SA7    A6+B1       DESTIN ADDR INC                         CGIA        4197     A
CGIA              LX6    AD.RBOCL                                            CGIA        4198     A
CGIA              SA2    S.                                                  CGIA        4199     A
CGIA              MX1    -AD.RBOCL                                           CGIA        4200     A
CGIA              BX2    -X1*X2                                              CGIA        4201     A
CGIA              IX7    X2+X7                                               CGIA        4202     A
CGIA              BX7    X0+X7                                               CGIA        4203     A
CGIA              SA7    A2+B1                                               CGIA        4204     A
CGIA              BX6    X6+X2                                               CGIA        4205     A
CGIA              SA6    A2                                                  CGIA        4206     A
CGIA              SA3    BCDC                                                CGIA        4207     A
CGIA              ZR     X3,REPI1    IF BCDC = 0                             CGIA        4208     A
CGIA              SA1    REPIA                                               CGIA        4209     A
CGIA              RJ     FMC         FORMAT *REPI* LINE                      CGIA        4210     A
CGIA              WRTEXT 0           AND LIST IT                             CGIA        4211     A
CGIA                                                                         CGIA        4212     A
CGIA      REPI1   RJ     FPU         FORCE UPPER                             CGIA        4213     A
CGIA              RJ     DTT         FLUSH OUT CURRENT TEXT TABLE            CGIA        4214     A
CGIA              MX7    -AD.RBOCL                                           CGIA        4215          I
          -CCGA005

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                BX3    -X7*X3                                          CGIA        4216          I
          -CCGA005
CGIA                SB3    LT.XTXT-2000B                                   CGIA        4217          I
          -CCGA005
CGIA                PX6    B3,X3                                           CGIA        4218          I
          -CCGA005
CGIA                SA6    TH          RESET *TH*                          CGIA        4219          I
          -CCGA005
CGIA                SETXTH X3          RESET *TH*                          CCGA005       33     A
CGIA      *                (X5) = SI WORD OR 0 IF NOT CALLED FROM USE    CCGA005       34     A
CGIA                SA2    REPIB                                           CCGA007       18     A
CGIA                MX4    -AD.RBL                                         CCGA007       19     A
CGIA                LX4    12                                              CCGA007       20     A
CGIA                BX6    X4*X2       CLEAR RB NUMBER FROM CL FIELD       CCGA007       21     A
CGIA                LX3    59-AD.CLP                                       CCGA007       22     A
CGIA                PL     X3,REPI2    IF NOT CONDITIONAL LOADING          CCGA007       23     A
CGIA                LX3    1+AD.CLP-AD.RBP+12                              CCGA007       24     A
CGIA                BX3    -X4*X3                                          CCGA007       25     A
CGIA                BX6    X3+X6       SET RB NUMBER IN CL FIELD           CCGA007       26     A
CGIA       REPI2    SA6    A2                                              CCGA007       27     A
CGIA                WRLGO  REPIB,3     REPI TABLE TO LGO                   CGIA        4220     A
CGIA                EQ     GNIW                                            CGIA        4221     A
CGIA                                                                       CGIA        4222     A
CGIA       REPIA    PSEUDO REPI,(C,C,C,C)                                  CGIA        4223     A
CGIA       REPIB    VFD    12/LT.REPI,12/2,36/1      XREP, I = 1           CGIA        4224     A
CGIA       S.       BSS    1           9/0,18/INC,33/RBOC[S.]              CGIA        4225     A
CGIA                BSS    1           15/RC,12/DLEN,33/RBOC[DESTIN]       CGIA        4226     A
CGIA       USE      SPACE  2,14                                            CGIA        4227     A
CGIA      **        USE    0,BN - SET NEW USE BLOCK FOR CODE PLACEMENT    CGIA        4228     A
CGIA                                                                       CGIA        4229     A
CGIA                PROCESS USE                                            CGIA        4230     A
CGIA                SX6    X5                                              CGIA        4231     A
CGIA                RJ     CUB         CHANGE BLOCKS                       CGIA        4232     A
CGIA                ZR     X7,GNIW     IF NO CHANGE                        CGIA        4233     A
CGIA                SA4    A5+B1                                           CCGA072       23     A
CGIA                BX6    X4-X5                                           CCGA072       24     A
CGIA                AX6    48                                              CCGA072       25     A
CGIA                ZR     X6,GNIW     SKIP LIST IF NEXT IS A USE PSEUDO   CCGA072       26     A
CGIA                SA2    AI                                              CGIA        4234     A
CGIA                SX3    B1                                              CGIA        4235     A
CGIA                LX2    -AD.LCMP                                        CGIA        4236     A
CGIA                BX4    X3*X2                                           CGIA        4237     A
CGIA                UX5    X5                                              CCGA005       35     A
CGIA                LX5    SI.CAP      SHIFT TO MAKE IH LISTED AS ARG1     CGIA        4238     A
CGIA                SA1    USEA+X4                                         CGIA        4239     A
CGIA                RJ     LPO         LIST PSEUDO                         CGIA        4240     A
CGIA                WRTEXT 0           LIST ADDRESS + LINE                 CGIA        4241     A
CGIA                EQ     GNIW                                            CGIA        4242     A
CGIA                                                                       CGIA        4243     A
CGIA       USEA     PSEUDO USE,B                                           CGIA        4244          I
          -CCGA005
CGIA       USEA     PSEUDO USE,(B,C)                                       CCGA005       36     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA              PSEUDO USELCM,B                                          CGIA        4245     A
CGIA      CUB     SPACE  3,14                                              CGIA        4246     A
CGIA     **       CUB - CHANGE USE BLOCK                                   CGIA        4247     A
CGIA     *                                                                 CGIA        4248     A
CGIA     *        ENTRY  (X6) = NUBO, NEW USE BLOCK ORIDNAL                CGIA        4249     A
CGIA     *                                                                 CGIA        4250     A
CGIA     *        EXIT   (X7) = 0 IF NO CHANGE                             CGIA        4251     A
CGIA     *                                                                 CGIA        4252     A
CGIA     *        PRESERVES  A0,A5, X0,X5                                  CGIA        4253     A
CGIA                                                                       CGIA        4254     A
CGIA      CUB     ROUTINE                                                  CGIA        4255     A
CGIA              SA4    UBO                                               CGIA        4256     A
CGIA              IX7    X4-X6                                             CGIA        4257     A
CGIA              SA6    A4+B1       UBO(2) = NUBO                         CGIA        4258     A
CGIA              ZR     X7,CUB      IF NUBO = UBO(1)  */ NEW = CURRENT    CGIA        4259     A
CGIA                                                                       CGIA        4260     A
CGIA              RJ     DTT         DUMP CURRENT TEXT TABLE               CGIA        4261     A
CGIA              SA4    UBO                                               CGIA        4262     A
CGIA              LX4    1                                                 CGIA        4263     A
CGIA              BX7    X3                                                CGIA        4264     A
CGIA              SX2    X4-1S16     I = 2*(UBO-1S15)                      CGIA        4265     A
CGIA              MI     X2,CUB1     IF I < 0      */ COMMON BLOCK         CGIA        4266     A
CGIA                                                                       CGIA        4267     A
CGIA              SA1    O.LBIT                                            CGIA        4268     A
CGIA              IX2    X1+X2                                             CGIA        4269     A
CGIA              SA6    X2+B1       LBIT(I+1) = CW INFO                   CGIA        4270     A
CGIA              SA7    X2          LBIT(I) = AI                          CGIA        4271     A
CGIA              EQ     CUB2                                              CGIA        4272     A
CGIA                                                                       CGIA        4273     A
CGIA      CUB1    SA1    O.CBT                                             CGIA        4274     A
CGIA              IX2    X1+X4                                             CGIA        4275     A
CGIA              SA7    X2          CBT(2*UBO) = AI                       CGIA        4276     A
CGIA                                                                       CGIA        4277     A
CGIA     *        SETUP AI, CW, TEXT HEADER FOR THE NEW BLOCK              CGIA        4278     A
CGIA                                                                       CGIA        4279     A
CGIA      CUB2    SA3    A4+B1                                             CGIA        4280     A
CGIA              BX6    X3                                                CGIA        4281     A
CGIA              SA6    A4          UBO(1) = UBO(2)                       CGIA        4282     A
CGIA              LX3    1                                                 CGIA        4283     A
CGIA              SX7    X3-1S16     I = 2*(UBO-1S15)                      CGIA        4284     A
CGIA              PL     X7,CUB3     IF I \ 0      */ A LOCAL BLOCK        CGIA        4285     A
CGIA                                                                       CGIA        4286     A
CGIA              SA2    O.CBT                                             CGIA        4287     A
CGIA              IX7    X2+X3       TBL = O.CBT + 2*UBO                   CGIA        4288     A
CGIA              MX6    0           V = 0                                 CGIA        4289     A
CGIA              BX2    X5                                                CCGA005       37     A
CGIA              LX2    -SI.CAP                                           CCGA005       38     A
CGIA              SX2    X2                                                CCGA005       39     A
CGIA              LX2    AD.CLP      CL = CA[SI]                           CCGA005       40     A
CGIA              EQ     CUB4                                              CGIA        4290     A
CGIA                                                                       CGIA        4291     A
CGIA      CUB3    SA2    O.LBIT                                            CGIA        4292     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                IX7    X2+X7       TBL = O.LBIT + I                          CGIA        4293     A
CGIA                SA3    X7+B1       LCW = LBIT(I+1)                           CGIA        4294     A
CGIA                MX4    -4                                                    CGIA        4295     A
CGIA                BX6    -X4*X3                                                CGIA        4296     A
CGIA                LX6    -4                                                    CGIA        4297     A
CGIA                SA6    TR          TR = SHIFT(LCW&17B,-4)                    CGIA        4298     A
CGIA                BX6    X4*X3       V = MASK(45) & LCM                        CGIA        4299     A
CGIA                MX2    0                                                     CCGA005       41     A
CGIA                                                                             CGIA        4300     A
CGIA       CUB4     SA3    X7                                                    CGIA        4301     A
CGIA                SA6    CW          CW = V                                    CGIA        4302     A
CGIA                BX6    X3                                                    CGIA        4303          I
           -CCGA005
CGIA                SA6    A6-B1       AI = TBL(1)                               CGIA        4304          I
           -CCGA005
CGIA                MX4    -AD.RBOCL                                             CGIA        4305          I
           -CCGA005
CGIA                SB3    LT.XTXT-2000B                                         CGIA        4306          I
           -CCGA005
CGIA                BX3    -X4*X3                                                CGIA        4307          I
           -CCGA005
CGIA                PX6    B3,X3                                                 CGIA        4308          I
           -CCGA005
CGIA                SA6    TH          TH = 12/LT.XTXT,12/,36/RBOC[AI]           CGIA        4309          I
           -CCGA005
CGIA                BX6    X3+X2                                                 CCGA005       42     A
CGIA                SA6    A6-B1       AI = TBL(1) ! CL                          CCGA005       43     A
CGIA                SETXTH X6          RESET *TH*                                CCGA005       44     A
CGIA                EQ     CUB                                                   CGIA        4310     A
CGIA       VFDP     SPACE  3,14                                                  CGIA        4311     A
CGIA     **         VFDP   BN,LEN AND 1 WORD OF LEFT JUSTIFIED DATA FOLLOWS      CGIA        4312     A
CGIA                                                                             CGIA        4313     A
CGIA                PROCESS VFDP                                                 CGIA        4314     A
CGIA                SA4    AI                                                    CGIA        4315     A
CGIA                MX1    -AD.RBOCL                                             CGIA        4316     A
CGIA                BX7    -X1*X4                                                CGIA        4317     A
CGIA                SX3    X5                                                    CGIA        4318     A
CGIA                LX3    42                                                    CGIA        4319     A
CGIA                BX7    X3+X7                                                 CGIA        4320     A
CGIA                AX6    SI.CAP                                                CGIA        4321     A
CGIA                LX6    36                                                    CGIA        4322     A
CGIA                BX7    X6+X7       18/LEN,6/BN,36/RBOC[AI]                   CGIA        4323     A
CGIA                SA7    VFDP+1                                                CGIA        4324     A
CGIA                UX0    X5                                                    CGIA        4325     A
CGIA                GNIW               GET DATA WORD                             CGIA        4326     A
CGIA                BX7    X5                                                    CGIA        4327     A
CGIA                SA7    VFDP+2                                                CGIA        4328     A
CGIA                SX1    X0+59                                                 CCGA067        5     A
CGIA                SX2    1S22/60+1                                             CCGA067        6     A
CGIA                IX3    X1*X2                                                 CCGA067        7     A
CGIA                AX3    22          N.DATA-WORDS = (LEN+59)/60                CCGA067        8     A
CGIA                SX6    X3-1                                                  CCGA067        9     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA          SA6    VFDPA       NUMBER OF WORDS REMAINING                CCGA067      10    A
CGIA          SX3    X3+1        NUMBER OF WORDS IN TABLE                 CCGA067      11    A
CGIA          SX7    LT.VFDP                                              CCGA067      12    A
CGIA          LX7    48                                                   CCGA067      13    A
CGIA          LX3    36                                                   CCGA067      14    A
CGIA          BX7    X7+X3                                                CCGA067      15    A
CGIA          SA7    VFDP                                                 CCGA067      16    A
CGIA          WRLGO  VFDP,3      VFDP TABLE TO LGO                        CGIA       4329    A
CGIA          SA2    BCDC                                                 CGIA       4330    A
CGIA          ZR     X2,GNIW     IF BCDC = 0                              CGIA       4331         I
        -CCGA067
CGIA          ZR     X2,VFDP1                                             CCGA067      17    A
CGIA                                                                      CGIA       4332    A
CGIA          LX0    -SI.CAP                                              CGIA       4333    A
CGIA          SX1    X0                                                   CGIA       4334    A
CGIA          SX5    1R,                                                  CGIA       4335    A
CGIA          RJ     CND         CONVERT BN AND ADD ,                     CGIA       4336    A
CGIA          SA6    LBUF+2                                               CGIA       4337    A
CGIA          LX0    SI.CAP                                               CGIA       4338    A
CGIA          SX1    X0                                                   CGIA       4339    A
CGIA          SX5    1R/                                                  CGIA       4340    A
CGIA          RJ     CND         CONVERT LEN/                             CGIA       4341    A
CGIA          SA6    A6+B1                                                CGIA       4342    A
CGIA          SA4    VFDP+2                                               CGIA       4343    A
CGIA          SB2    X0                                                   CGIA       4344         I
        -CCGA067
CGIA          LX4    B2                                                   CGIA       4345         I
        -CCGA067
CGIA          RJ     CFW         CONVERT DATA                             CGIA       4346    A
CGIA          SA7    A6+B1       UPPER                                    CGIA       4347    A
CGIA          SA6    A7+B1       LOWER                                    CGIA       4348    A
CGIA          SA1    =1H                                                  CGIA       4349    A
CGIA          SA2    =8LVFDP                                              CGIA       4350    A
CGIA          SB4    12          UBC = 12                                 CGIA       4351    A
CGIA          BX7    X1                                                   CGIA       4352    A
CGIA          SA7    LBUF                                                 CGIA       4353    A
CGIA          BX7    X2                                                   CGIA       4354    A
CGIA          MX0    42                                                   CGIA       4355    A
CGIA          ECHO   4,Z=(LBUF+2,A3+B1)                                   CGIA       4356    A
CGIA          SA3    Z                                                    CGIA       4357    A
CGIA          BX6    X0*X3                                                CGIA       4358    A
CGIA          SB3    X3                                                   CGIA       4359    A
CGIA          RJ     ADC                                                  CGIA       4360    A
CGIA          SB3    60                                                   CGIA       4361    A
CGIA          DUP    2,3                                                  CGIA       4362    A
CGIA          SA3    A3+B1                                                CGIA       4363    A
CGIA          BX6    X3                                                   CGIA       4364    A
CGIA          RJ     ADC                                                  CGIA       4365    A
CGIA          R=     X6,1RB                                               CGIA       4366    A
CGIA          LX6    -6                                                   CGIA       4367    A
CGIA          SB3    6                                                    CGIA       4368    A
CGIA          RJ     ADC         ADD FINAL B                              CGIA       4369    A

        0         1         2         3         4         5         6         7         8
        12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                RJ     TSB                                                  CGIA        4370     A
CGIA                WRTEXT 0            LIST THE LINE                           CGIA        4371     A
CGIA                EQ     GNIW                                                 CGIA        4372          I
          -CCGA067
CGIA                                                                            CCGA067       18     A
CGIA       VFDP1    SA2    VFDPA                                                CCGA067       19     A
CGIA                ZR     X2,GNIW      IF NO MORE TEXT                         CCGA067       20     A
CGIA                SX7    X2-1                                                 CCGA067       21     A
CGIA                SA7    A2                                                   CCGA067       22     A
CGIA                GNIW                GET DATA WORD                           CCGA067       23     A
CGIA                BX7    X5                                                   CCGA067       24     A
CGIA                SA7    VFDP+2                                               CCGA067       25     A
CGIA                BX4    X5                                                   CCGA067       26     A
CGIA                RJ     CFW          CONVERT FULL WORD                       CCGA067       27     A
CGIA                SA7    LBUF+2       STORE UPPER                             CCGA067       28     A
CGIA                BX7    X6                                                   CCGA067       29     A
CGIA                SB4    B0           UBC = 0                                 CCGA067       30     A
CGIA                SA1    =1H                                                  CCGA067       31     A
CGIA                BX6    X1                                                   CCGA067       32     A
CGIA                SA6    A7-B1        BLANK OPCODE FIELD                      CCGA067       33     A
CGIA                SA6    A6-B1                                                CCGA067       34     A
CGIA                RJ     TSB          TERMINATE STRING AND PRINT LINE         CCGA067       35     A
CGIA                WRTEXT 0                                                    CCGA067       36     A
CGIA                WRLGO  VFDP+2,1     DATA WORD TO LGO                        CCGA067       37     A
CGIA                EQ     VFDP1                                                CCGA067       38     A
CGIA                                                                            CGIA        4373     A
CGIA       VFDP     VFD    12/LT.VFDP,12/2,36/0                                 CGIA        4374     A
CGIA                BSS    2            18/LEN,6/BN,36/RBOC AND DATA WORD       CGIA        4375     A
CGIA       VFDPA    BSS    1                                                    CCGA067       39     A
CGIA       CND      SPACE  3,14                                                 CGIA        4376     A
CGIA     **         CND - CONVERT NUMBER TO DECIMAL AND ADD SEPERATOR           CGIA        4377     A
CGIA     *                                                                      CGIA        4378     A
CGIA     *          ENTRY  (X1) = BINARY NUMBER                                 CGIA        4379     A
CGIA     *                 (X5) = 1R_SEP-CHAR                                   CGIA        4380     A
CGIA     *                                                                      CGIA        4381     A
CGIA     *          EXIT   (X6) = 42/0L_NUM_SEP-CHAR,18/N.BITS                  CGIA        4382     A
CGIA                                                                            CGIA        4383     A
CGIA       CND      ROUTINE                                                     CGIA        4384     A
CGIA                CALL   CDD=                                                 CGIA        4385     A
CGIA                SB3    B2-B1                                                CGIA        4386     A
CGIA                MX2    B3                                                   CGIA        4387     A
CGIA                BX6    X2*X4        REMOVE TRAILING BLANKS                  CGIA        4388     A
CGIA                SB2    B2+6                                                 CGIA        4389     A
CGIA                SB3    B2-60                                                CGIA        4390     A
CGIA                LX5    -B3                                                  CGIA        4391     A
CGIA                BX6    X6+X5        ADD SEP CHAR                            CGIA        4392     A
CGIA                SX2    B2                                                   CGIA        4393     A
CGIA                BX6    X6+X2        AND BIT COUNT                           CGIA        4394     A
CGIA                EQ     CND                                                  CGIA        4395     A
CGIA                SPACE  3                                                    CGIA        4396     A
CGIA       .TS      IFEQ   HC.TS,0                                              CGIA        4397          I
          -CCG16

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA      CGEP     TITLE  CGEP - CODE GENERATOR END PROCESSOR ROUTINES                CGIA        4398          I
          -CCG16
CGIA      **       CGEP - CODE GENERATOR END PROCESSOR                                 CGIA        4399          I
          -CCG16
CGIA      *                                                                            CGIA        4400          I
          -CCG16
CGIA      *        *CGEP* CONTAINS ROUTINE TO DO FINAL ADDRESS DEFINITION,             CGIA        4401          I
          -CCG16
CGIA      *        BLOCK LENGTH CALCULATION, BLOCK SWITCHING, ETC.                     CGIA        4402          I
          -CCG16
CGIA      *                                                                            CGIA        4403          I
          -CCG16
CGIA      *        NOTE THAT SINCE THE CODE IN THE END PROCESSOR IS USED BY THE        CGIA        4404          I
          -CCG16
CGIA      *        ASSEMBLER FOR SCRATCH STORAGE IT MAY NOT BE ACCESSED BY IT.         CGIA        4405          I
          -CCG16
CGIA                                                                                   CGIA        4406          I
          -CCG16
CGIA      N$SLBT   ENTRY. 0                                                            CGIA        4407          I
          -CCG16
CGIA               USE    WII                                                          CGIA        4408          I
          -CCG16
CGIA                                                                                   CGIA        4409          I
          -CCG16
CGIA               TABLES (CUT,CVT,PIT)                                                CGIA        4410          I
          -CCG16
CGIA                                                                                   CGIA        4411          I
          -CCG16
CGIA      N$IT     ENTRY. 0                                                            CGIA        4412          I
          -CCG16
CGIA      N$OT     ENTRY. 0                                                            CGIA        4413          I
          -CCG16
CGIA                                                                                   CGIA        4414          I
          -CCG16
CGIA      CC$LBO   ENTRY. 0          ORDINAL OF LOCAL BLOCK BEING PROCESSED           CGIA        4415          I
          -CCG16
CGIA      CC$PC    ENTRY. 0          PARCEL COUNTER ( 0 - 3 )                         CGIA        4416          I
          -CCG16
CGIA      CC$BLEN  ENTRY. 0          LENGTH OF CURRENT BLOCK                          CGIA        4417          I
          -CCG16
CGIA                                                                                   CGIA        4418          I
          -CCG16
CGIA               IFNE   HC.FPAS,0,1                                                  CGIA        4419          I
          -CCG16
CGIA      LSR      DATA   0          LAST SUB REF  12/P(FPO),30/,18/BLEN              CGIA        4420          I
          -CCG16
CGIA                                                                                   CGIA        4421          I
          -CCG16
CGIA               QUAL   WII                                                          CGIA        4422          I
          -CCG16
CGIA      CG$RBT   SPACE  2,14                                                         CGIA        4423          I
          -CCG16

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA     **        CG$RBT - RELOCATE BLOCK TABLE                                    CGIA        4424          I
         -CCG16
CGIA                                                                                CGIA        4425          I
         -CCG16
CGIA      CG$RBT   ROUTINE                                                          CGIA        4426          I
         -CCG16
CGIA               SA3    CC$PC                                                     CGIA        4427          I
         -CCG16
CGIA               SA4    CC$BLEN                                                   CGIA        4428          I
         -CCG16
CGIA               SA1    O.LBT                                                     CGIA        4429          I
         -CCG16
CGIA               SA2    CC$LBO                                                    CGIA        4430          I
         -CCG16
CGIA               LX3    18                                                        CGIA        4431          I
         -CCG16
CGIA               BX6    X3+X4                                                     CGIA        4432          I
         -CCG16
CGIA               SB2    X2                                                        CGIA        4433          I
         -CCG16
CGIA               SA6    X1+B2       LBT(LBO) = (PC,BLEN)                          CGIA        4434          I
         -CCG16
CGIA               SA3    L.LBT                                                     CGIA        4435          I
         -CCG16
CGIA               SB2    B0          I = 0                                         CGIA        4436          I
         -CCG16
CGIA               SB3    X3          N = L.LBT                                     CGIA        4437          I
         -CCG16
CGIA               MX0    42                                                        CGIA        4438          I
         -CCG16
CGIA               MX7    0           S = 0                                         CGIA        4439          I
         -CCG16
CGIA                                                                                CGIA        4440          I
         -CCG16
CGIA      RBT1     SA3    X1+B2                                                     CGIA        4441          I
         -CCG16
CGIA               BX5    X0*X3                                                     CGIA        4442          I
         -CCG16
CGIA               ZR     X5,RBT2     IF PC[LBT(I)] = 0                             CGIA        4443          I
         -CCG16
CGIA               SX3    X3+1        BLEN[LBT(I)] = BLEN[LBT(I)] + 1               CGIA        4444          I
         -CCG16
CGIA      RBT2     SA7    A3          LBT(I) = S    */ FWA                          CGIA        4445          I
         -CCG16
CGIA               IX7    X7+X3       S = S + BLEN[LBT(I)]                          CGIA        4446          I
         -CCG16
CGIA               SB2    B2+1        I = I + 1                                     CGIA        4447          I
         -CCG16
CGIA               LT     B2,B3,RBT1                                                CGIA        4448          I
         -CCG16
CGIA                                                                                CGIA        4449          I
         -CCG16

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                SA7    N$SLBT                                                CGIA        4450          I
          -CCG16
CGIA                EQ     CG$RBT                                                CGIA        4451          I
          -CCG16
CGIA       CG$CUB   TITLE  CG$CUB - CHANGE USE BLOCK                             CGIA        4452          I
          -CCG16
CGIA      **        CG$CUB - CHANGE USE BLOCK                                    CGIA        4453          I
          -CCG16
CGIA      *                                                                      CGIA        4454          I
          -CCG16
CGIA      *         ENTRY  (X1) = *LBT* ORDINAL OF NEW USE BLOCK                 CGIA        4455          I
          -CCG16
CGIA                                                                             CGIA        4456          I
          -CCG16
CGIA       CG$CUB   ROUTINE                                                      CGIA        4457          I
          -CCG16
CGIA                SETB1                                                        CGIA        4458          I
          -CCG16
CGIA                SA2    CC$LBO      OBO = CC$LBO                              CGIA        4459          I
          -CCG16
CGIA                IX7    X1-X2                                                 CGIA        4460          I
          -CCG16
CGIA                BX6    X1                                                    CGIA        4461          I
          -CCG16
CGIA                ZR     X7,CG$CUB   IF NBO = OBO                             CGIA        4462          I
          -CCG16
CGIA                                                                             CGIA        4463          I
          -CCG16
CGIA                SA6    A2          CC$LBO = NBO                              CGIA        4464          I
          -CCG16
CGIA                SA3    CC$PC                                                 CGIA        4465          I
          -CCG16
CGIA                SA4    CC$BLEN                                               CGIA        4466          I
          -CCG16
CGIA                LX3    18                                                    CGIA        4467          I
          -CCG16
CGIA                BX6    X3+X4                                                 CGIA        4468          I
          -CCG16
CGIA                SA5    O.LBT                                                 CGIA        4469          I
          -CCG16
CGIA                SB3    X5                                                    CGIA        4470          I
          -CCG16
CGIA                SA6    B3+X2       LBT(OBO) = SHIFT(PC,18) ! BLEN            CGIA        4471          I
          -CCG16
CGIA                SA5    B3+X1                                                 CGIA        4472          I
          -CCG16
CGIA                SX6    X5                                                    CGIA        4473          I
          -CCG16
CGIA                SA6    A4                                                    CGIA        4474          I
          -CCG16
CGIA                AX5    18          SET NEW PARCEL AND BLEN                   CGIA        4475          I
          -CCG16

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                SX6    X5                                                    CGIA        4476          I
          -CCG16
CGIA                SA6    A3                                                    CGIA        4477          I
          -CCG16
CGIA                SX6    X1+1S15                                               CGIA        4478          I
          -CCG16
CGIA                WRITEP USE         OUTPUT *USE* TO SLIST                     CGIA        4479          I
          -CCG16
CGIA                EQ     CG$CUB                                                CGIA        4480          I
          -CCG16
CGIA       FSU#     SPACE  3,14                                                  CGIA        4481          I
          -CCG16
CGIA      **        FSU# - FORCE NEXT SEQUENCE UPPER                             CGIA        4482          I
          -CCG16
CGIA                                                                             CGIA        4483          I
          -CCG16
CGIA       FSU#     ROUTINE                                                      CGIA        4484          I
          -CCG16
CGIA                SA1    CC$PC                                                 CGIA        4485          I
          -CCG16
CGIA                ZR     X1,FSU#     IF PARCEL = 0                            CGIA        4486          I
          -CCG16
CGIA                MX7    0                                                     CGIA        4487          I
          -CCG16
CGIA                SA7    A1          PARCEL = 0                               CGIA        4488          I
          -CCG16
CGIA                SA2    CC$BLEN                                               CGIA        4489          I
          -CCG16
CGIA                SX6    X2+B1       BLEN = BLEN + 1                          CGIA        4490          I
          -CCG16
CGIA                SA6    A2                                                    CGIA        4491          I
          -CCG16
CGIA                SA6    FU          FU = 1                                   CGIA        4492          I
          -CCG16
CGIA                EQ     FSU#                                                  CGIA        4493          I
          -CCG16
CGIA                                                                             CGIA        4494          I
          -CCG16
CGIA       FU       DATA   0           "0 IF TO FORCE UPPER                     CGIA        4495          I
          -CCG16
CGIA       CG$EP    TITLE  CG$EP - CCG END PROCESSOR                             CGIA        4496          I
          -CCG16
CGIA      **        CG$EP - CODE GENERATOR END PROCESSOR                         CGIA        4497          I
          -CCG16
CGIA      *         FUNCTIONS -                                                  CGIA        4498          I
          -CCG16
CGIA      *         SETUP *CVT* ORDINAL TABLE FOR *CGIA*                         CGIA        4499          I
          -CCG16
CGIA      *         OUPUT TABLE OF CONSTANT LITERALS AS - * CON. BSS 0 FOLLOWED  CGIA        4500          I
          -CCG16
CGIA      *         BY DATA WORDS.                                               CGIA        4501          I
          -CCG16

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA     *         DEFINE ADDRESS"S AND ALLOCATE STORAGE FOR OTHER SPECIAL SYMBOLS.      CGIA        4502          I
         -CCG16
CGIA     *                                                                              CGIA        4503          I
         -CCG16
CGIA     *         ENTRY  CVT - CONSTANT VALUE TABLE                                    CGIA        4504          I
         -CCG16
CGIA     *                CUT - CONSTANT USE TABLE ENTRIES ARE 0 OR 1 FOR UNUSED       CGIA        4505          I
         -CCG16
CGIA     *                   OR USED.                                                  CGIA        4506          I
         -CCG16
CGIA     *                N$IT, N$OT = NUMBER OF IT."S USED, ETC                       CGIA        4507          I
         -CCG16
CGIA     *                                                                              CGIA        4508          I
         -CCG16
CGIA     *         EXIT   CVT REFORMATTED AS 60/ ORDINAL OF CONSTANT VALUE IN           CGIA        4509          I
         -CCG16
CGIA     *                   CON. ARRAY.                                               CGIA        4510          I
         -CCG16
CGIA     *                ADDRESS OF SPECIAL SYMBOLS ( IT., OT., CON. , ETC )          CGIA        4511          I
         -CCG16
CGIA     *                   DEFINED AND OUTPUT TO *SLIST* FILE.                       CGIA        4512          I
         -CCG16
CGIA                                                                                    CGIA        4513          I
         -CCG16
CGIA      CG$EP    ROUTINE                                                              CGIA        4514          I
         -CCG16
CGIA               SETB1                                                                CGIA        4515          I
         -CCG16
CGIA                                                                                    CGIA        4516          I
         -CCG16
CGIA     *         REFORMAT *CUT* AS ORDINALS TO CON. ARRAY, SQUEEZE UNUSED             CGIA        4517          I
         -CCG16
CGIA     *         ENTRIES OUT OF *CVT*.                                                CGIA        4518          I
         -CCG16
CGIA                                                                                    CGIA        4519          I
         -CCG16
CGIA               ADDWRD CUT,0                                                         CGIA        4520          I
         -CCG16
CGIA               SA1    O.CVT                                                         CGIA        4521          I
         -CCG16
CGIA               SB2    B0          I = 0                                             CGIA        4522          I
         -CCG16
CGIA               SB3    X3          N = L.CUT                                         CGIA        4523          I
         -CCG16
CGIA               SB4    X1                      (B4) = CVT                            CGIA        4524          I
         -CCG16
CGIA               SB5    X2                      (B5) = CUT                            CGIA        4525          I
         -CCG16
CGIA               SX7    0           J = 0                                             CGIA        4526          I
         -CCG16
CGIA                                                                                    CGIA        4527          I
         -CCG16

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA      EP1      SA5    B5+B2                                                    CGIA        4528          I
          -CCG16
CGIA               ZR     X5,EP2      IF CUT(I) = 0 */ CON NOT REFERENCED          CGIA        4529          I
          -CCG16
CGIA                                                                               CGIA        4530          I
          -CCG16
CGIA               SA7    A5          CUT(I) = J    */ SET ORDINAL                 CGIA        4531          I
          -CCG16
CGIA               SA4    B4+B2                                                    CGIA        4532          I
          -CCG16
CGIA               BX6    X4          CVT(J) = CVT(I)  */ MOVE TO FINAL POSITION   CGIA        4533          I
          -CCG16
CGIA               SA6    B4+X7                                                    CGIA        4534          I
          -CCG16
CGIA               SX7    X7+B1       J = J + 1                                    CGIA        4535          I
          -CCG16
CGIA                                                                               CGIA        4536          I
          -CCG16
CGIA      EP2      SB2    B2+B1       I = I + 1                                    CGIA        4537          I
          -CCG16
CGIA               LT     B2,B3,EP1   IF I < N                                     CGIA        4538          I
          -CCG16
CGIA                                                                               CGIA        4539          I
          -CCG16
CGIA               BX5    X7                                                       CGIA        4540          I
          -CCG16
CGIA               ZR     X7,EP3      IF J = 0                                     CGIA        4541          I
          -CCG16
CGIA      *                                                                        CGIA        4542          I
          -CCG16
CGIA      *        DEFINE ADDRESS OF CON. AND OUTPUT TABLE OF DATA WORDS           CGIA        4543          I
          -CCG16
CGIA                                                                               CGIA        4544          I
          -CCG16
CGIA               SA1    =XS$CON                                                  CGIA        4545          I
          -CCG16
CGIA               SX2    B0                                                       CGIA        4546          I
          -CCG16
CGIA               RJ     CG$DSA      DEFINE CON.                                  CGIA        4547          I
          -CCG16
CGIA               BX6    X5                                                       CGIA        4548          I
          -CCG16
CGIA               LX6    SI.CAP                                                   CGIA        4549          I
          -CCG16
CGIA               WRITEP DATA        OUTPUT DATA HEADER WORD                      CGIA        4550          I
          -CCG16
CGIA               SA4    O.CVT                                                    CGIA        4551          I
          -CCG16
CGIA               WRITEW F.SLST,X4,X5      OUTPUT DATA WORDS                      CGIA        4552          I
          -CCG16
CGIA                                                                               CGIA        4553          I
          -CCG16

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA     *         DEFINE ADDRESS"S OF SPECIAL SYMBOLS                                CGIA     4554     I
         -CCG16
CGIA                                                                                 CGIA     4555     I
         -CCG16
CGIA      EP3      SX6    0                                                           CGIA     4556     I
         -CCG16
CGIA                SA6    L.CVT       L.CVT = 0                                       CGIA     4557     I
         -CCG16
CGIA      .PLI     IFNE   HC.ID,3                                                     CGIA     4558     I
         -CCG16
CGIA                ECHO   4,Z=(IT,OT)                                                 CGIA     4559     I
         -CCG16
CGIA                SA1    =XS$Z                                                       CGIA     4560     I
         -CCG16
CGIA                SA2    N$Z                                                         CGIA     4561     I
         -CCG16
CGIA     +         ZR     X2,*+1                                                      CGIA     4562     I
         -CCG16
CGIA                RJ     CG$DSA                                                      CGIA     4563     I
         -CCG16
CGIA      .PLI     ENDIF                                                              CGIA     4564     I
         -CCG16
CGIA                                                                                 CGIA     4565     I
         -CCG16
CGIA                EQ     CG$EP                                                       CGIA     4566     I
         -CCG16
CGIA      CG$DSA   SPACE  3,14                                                        CGIA     4567     I
         -CCG16
CGIA     **        CG$DSA - DEFINE SYMBOL ADDRESS AND OUTPUT STORAGE FOR IT           CGIA     4568     I
         -CCG16
CGIA     *                                                                           CGIA     4569     I
         -CCG16
CGIA     *         ENTRY  (X1) = IH, SYMTAB ORDINAL                                   CGIA     4570     I
         -CCG16
CGIA     *                (X2) = WDS , NUMBER OF WORDS RESERVED FOR THIS SYMBOL      CGIA     4571     I
         -CCG16
CGIA     *                                                                           CGIA     4572     I
         -CCG16
CGIA     *         PRESERVES A0, X0, A5, X5                                           CGIA     4573     I
         -CCG16
CGIA                                                                                 CGIA     4574     I
         -CCG16
CGIA      CG$DSA   ROUTINE                                                            CGIA     4575     I
         -CCG16
CGIA                SA4    CC$PC                                                       CGIA     4576     I
         -CCG16
CGIA                SETB1                                                              CGIA     4577     I
         -CCG16
CGIA                SA3    CC$BLEN                                                     CGIA     4578     I
         -CCG16
CGIA                SX6    B1                                                          CGIA     4579     I
         -CCG16

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                LX6    WC.RLP                                                  CGIA        4580          I
          -CCG16
CGIA                ZR     X4,DSA1     IF PARCEL = 0                               CGIA        4581          I
          -CCG16
CGIA                MX7    0                                                       CGIA        4582          I
          -CCG16
CGIA                SA7    A4          PARCEL = 0                                  CGIA        4583          I
          -CCG16
CGIA                SX3    X3+1        BLEN = BLEN + 1                             CGIA        4584          I
          -CCG16
CGIA       DSA1     IX7    X3+X2       RA = BLEN;  BLEN = BLEN + WDS               CGIA        4585          I
          -CCG16
CGIA                SA4    CC$LBO                                                  CGIA        4586          I
          -CCG16
CGIA                SA7    A3                                                      CGIA        4587          I
          -CCG16
CGIA                ZR     X1,DSA2     IF IH = 0                                   CGIA        4588          I
          -CCG16
CGIA                LX4    WC.RBP                                                  CGIA        4589          I
          -CCG16
CGIA                BX4    X4+X3                                                   CGIA        4590          I
          -CCG16
CGIA                SA3    O.SYM                                                   CGIA        4591          I
          -CCG16
CGIA                BX6    X6+X4       ADW = WC(1,0,CG.LBO,RA)                     CGIA        4592          I
          -CCG16
CGIA                SB2    X3+2                                                    CGIA        4593          I
          -CCG16
CGIA                LX7    B1,X1                                                   CGIA        4594          I
          -CCG16
CGIA                IX7    X1+X7                                                   CGIA        4595          I
          -CCG16
CGIA                SA6    B2+X7       SYM(3*IH+2) = ADW                           CGIA        4596          I
          -CCG16
CGIA       DSA2     LX2    R1.CAP                                                  CGIA        4597          I
          -CCG16
CGIA                BX6    X2+X1                                                   CGIA        4598          I
          -CCG16
CGIA                WRITEP LAB         OUTPUT * IH BSS WDS * TO SLIST              CGIA        4599          I
          -CCG16
CGIA                EQ     CG$DSA                                                  CGIA        4600          I
          -CCG16
CGIA        WII     TITLE  WII - WRITE ISSUED INSTRUCTIONS TO *SLIST*              CGIA        4601          I
          -CCG16
CGIA      **        WII - WRITE ISSUED INSTRUCTIONS IN *SI.* FORMAT TO *SLIST*.    CGIA        4602          I
          -CCG16
CGIA      *         FUNCTIONS -                                                    CGIA        4603          I
          -CCG16
CGIA      *                MAINTAIN BLOCK LENGTH AND PARCEL COUNTER                CGIA        4604          I
          -CCG16
CGIA      *                ADJUST SPECIAL INSTRUCTIONS ( TLD, TST, ETC. )          CGIA        4605          I
          -CCG16

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA     *                SET *MAT* BIT IN WORDB OF *SYM*                                CGIA        4606          I
         -CCG16
CGIA     *                DEFINE ADDRESS"S OF TRANSFER LABELS                            CGIA        4607          I
         -CCG16
CGIA                                                                                      CGIA        4608          I
         -CCG16
CGIA      SZ       MICRO  1,, B2      INSTRUCTION SIZE IN PARCELS                         CGIA        4609          I
         -CCG16
CGIA      PC       MICRO  1,, B5      PARCEL - 4                                          CGIA        4610          I
         -CCG16
CGIA      BL       MICRO  1,, B7      BLOCK LENGTH                                        CGIA        4611          I
         -CCG16
CGIA                                                                                      CGIA        4612          I
         -CCG16
CGIA      WII#     ROUTINE                                                                CGIA        4613          I
         -CCG16
CGIA               TRACE  WII,PIT                                                         CGIA        4614          I
         -CCG16
CGIA               SA5    O.PIT                                                           CGIA        4615          I
         -CCG16
CGIA               WRITEW F.SLST,X5,1       OUTPUT *BOS* TO SLIST                         CGIA        4616          I
         -CCG16
CGIA               SA1    FU                                                              CGIA        4617          I
         -CCG16
CGIA               ZR     X1,WII0     IF FU = 0     */ NO FORCE UPPER                      CGIA        4618          I
         -CCG16
CGIA               MX6    0           FU = 0                                              CGIA        4619          I
         -CCG16
CGIA               SA6    A1                                                              CGIA        4620          I
         -CCG16
CGIA               WRITEP BSS         OUTPUT FORCE UPPER (  BSS 0 )                       CGIA        4621          I
         -CCG16
CGIA                                                                                      CGIA        4622          I
         -CCG16
CGIA      WII0     SA5    X5          PI = [O.PIT]                                        CGIA        4623          I
         -CCG16
CGIA               SA2    O.SYM                                                           CGIA        4624          I
         -CCG16
CGIA               SB6    X2+B1                    (B6) = SYM + 1                         CGIA        4625          I
         -CCG16
CGIA               SA4    CC$PC                                                           CGIA        4626          I
         -CCG16
CGIA               SA3    CC$BLEN                                                         CGIA        4627          I
         -CCG16
CGIA               S"BL"  X3                                                              CGIA        4628          I
         -CCG16
CGIA               S"PC"  X4-4                                                            CGIA        4629          I
         -CCG16
CGIA               S"SZ"  B0          SZ = 0                                              CGIA        4630          I
         -CCG16
CGIA               SA2    O.CUT                                                           CGIA        4631          I
         -CCG16

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                MX0    -6                                                    CGIA        4632          I
          -CCG16
CGIA                SA0    X2                        (A0) = CUT                  CGIA        4633          I
          -CCG16
CGIA                                                                             CGIA        4634          I
          -CCG16
CGIA      *         RETURN POINT FOR PROCESSING OF MOST INSTRUCTIONS             CGIA        4635          I
          -CCG16
CGIA                                                                             CGIA        4636          I
          -CCG16
CGIA       WII1     S"PC"  "PC"+"SZ"   PC = PC + SZ                             CGIA        4637          I
          -CCG16
CGIA                SA5    A5+B1       PI = PI + 1                              CGIA        4638          I
          -CCG16
CGIA                LT     "PC",WII2   IF PC < 4                                CGIA        4639          I
          -CCG16
CGIA                                                                             CGIA        4640          I
          -CCG16
CGIA                SB3    "PC"        S = PC                                   CGIA        4641          I
          -CCG16
CGIA                S"PC"  -4          PC = 0                                   CGIA        4642          I
          -CCG16
CGIA                S"BL"  "BL"+B1     BLEN = BLEN + 1                          CGIA        4643          I
          -CCG16
CGIA                ZR     B3,WII2     IF S = 0      */ NO SIZE FAULT           CGIA        4644          I
          -CCG16
CGIA                                                                             CGIA        4645          I
          -CCG16
CGIA                S"PC"  "PC"+"SZ"   PC = SZ                                  CGIA        4646          I
          -CCG16
CGIA                                                                             CGIA        4647          I
          -CCG16
CGIA      *         SETUP REGISTERS, JUMP TO INSTRUCTION PROCESSOR               CGIA        4648          I
          -CCG16
CGIA                                                                             CGIA        4649          I
          -CCG16
CGIA       WII2     UX7    B3,X5                                                CGIA        4650          I
          -CCG16
CGIA                SA3    OC.AT+B3                                             CGIA        4651          I
          -CCG16
CGIA                S"SZ"  B1          SZ = 1                                   CGIA        4652          I
          -CCG16
CGIA                AX3    18                                                   CGIA        4653          I
          -CCG16
CGIA                SB4    X3                                                   CGIA        4654          I
          -CCG16
CGIA                JP     B4+WII1     JUMP TO PROCESSOR                        CGIA        4655          I
          -CCG16
CGIA       NOP      SPACE  2,14                                                 CGIA        4656          I
          -CCG16
CGIA      **        NOP - TERMINATE PROCESSING                                   CGIA        4657          I
          -CCG16

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                                                                                    CGIA        4658          I
          -CCG16
CGIA                PROCESS NOP                                                         CGIA        4659          I
          -CCG16
CGIA                SX6     "PC"+4                                                      CGIA        4660          I
          -CCG16
CGIA                SA6     CC$PC                                                       CGIA        4661          I
          -CCG16
CGIA                SX7     "BL"+                                                       CGIA        4662          I
          -CCG16
CGIA                SA7     CC$BLEN                                                     CGIA        4663          I
          -CCG16
CGIA                SA4     O.PIT                                                       CGIA        4664          I
          -CCG16
CGIA                SA3     L.PIT                                                       CGIA        4665          I
          -CCG16
CGIA                MX7     0                                                           CGIA        4666          I
          -CCG16
CGIA                SA7     A3          L.PIT = 0                                       CGIA        4667          I
          -CCG16
CGIA                WRITEW F.SLST,X4+B1,X3-2 WRITE REST OF *PIT* TO SLIST               CGIA        4668          I
          -CCG16
CGIA                EQ      WII#                                                        CGIA        4669          I
          -CCG16
CGIA       TLD      SPACE   3,14                                                        CGIA        4670          I
          -CCG16
CGIA      **        TLD, TST - ADJUST *CA* TO REFLECT ASSIGNMENT MADE BY *GPO*          CGIA        4671          I
          -CCG16
CGIA                                                                                    CGIA        4672          I
          -CCG16
CGIA                PROCESS (TLD,TST)                                                   CGIA        4673          I
          -FCC1842X -CCG16
CGIA       TLD      PROCESS (TLD,TST)                                                   FCC1842X       1          I
          -CCG16
CGIA                LX7     -SI.RJP                                                     CGIA        4674          I
          -CCG16
CGIA                BX6     -X0*X7                                                      CGIA        4675          I
          -CCG16
CGIA                NZ      X6,WII1     IF RJ[PI] " 0 */ CONVERTED TO A SHORT INSTRUCTION   CGIA    4676          I
          -CCG16
CGIA                                                                                    CGIA        4677          I
          -CCG16
CGIA      *         SEARCH *TET* TO FIND LOCATION IN IT. BLOCK ASSIGNED                 CGIA        4678          I
          -CCG16
CGIA      *         TO THIS TLD/TST .                                                   CGIA        4679          I
          -CCG16
CGIA                                                                                    CGIA        4680          I
          -CCG16
CGIA                SA1     =XO$TET                                                     CGIA        4681          I
          -CCG16
CGIA                SB3     59-T.FAP                                                    CGIA        4682          I
          -CCG16

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                SB2    X1                        (B2) = TET                          CGIA        4683          I
          -CCG16
CGIA                LX7    SI.RJP-SI.CAP                                                 CGIA        4684          I
          -CCG16
CGIA                SB4    59-T.EQVP                                                     CGIA        4685          I
          -CCG16
CGIA                                                                                     CGIA        4686          I
          -CCG16
CGIA       TLD1     SA1    B2+X7       TI = TET(C)                                       CGIA        4687          I
          -CCG16
CGIA                LX6    B3,X1                                                         CGIA        4688          I
          -CCG16
CGIA                SX7    X1          C = CA[TI]                                        CGIA        4689          I
          -CCG16
CGIA                LX2    B4,X1                                                         CGIA        4690          I
          -CCG16
CGIA                MI     X6,TLD3     IF FA[TI]     */ FINAL ASSIGNMENT MADE            CGIA        4691          I
          -CCG16
CGIA                MI     X2,TLD1     IF EQV[TI]    */ EQUIVALENT TO ANOTHER            CGIA        4692          I
          -CCG16
CGIA                                                                                     CGIA        4693          I
          -CCG16
CGIA      *         CA NOT ASSIGNED, MUST BE OPT=1, GET NEXT CELL AND SAVE               CGIA        4694          I
          -CCG16
CGIA      *         VALUE IN *CA* FIELD OF *TET* ENTRY.                                  CGIA        4695          I
          -CCG16
CGIA                                                                                     CGIA        4696          I
          -CCG16
CGIA                SA2    =XN.GT#                                                       CGIA        4697          I
          -CCG16
CGIA                BX7    X2          C = N.GT                                          CGIA        4698          I
          -CCG16
CGIA                MX3    -T.CAL                                                        CGIA        4699          I
          -CCG16
CGIA                SX6    X2+B1       N.GT = N.GT + 1                                   CGIA        4700          I
          -CCG16
CGIA                SA6    A2                                                            CGIA        4701          I
          -CCG16
CGIA                SA4    =XN$IT                                                        CGIA        4702          I
          -CCG16
CGIA                SX2    B1                                                            CGIA        4703          I
          -CCG16
CGIA                BX1    X3*X1                                                         CGIA        4704          I
          -CCG16
CGIA                BX1    X1+X7       CA[TI] = C                                        CGIA        4705          I
          -CCG16
CGIA                LX2    T.FAP                                                         CGIA        4706          I
          -CCG16
CGIA                BX6    X1+X2       FA[TI] = 1                                        CGIA        4707          I
          -CCG16
CGIA                SA6    A1                                                            CGIA        4708          I
          -CCG16

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                SX6    X7+B1                                                CGIA        4709          I
          -CCG16
CGIA                IX4    X4-X6                                                CGIA        4710          I
          -CCG16
CGIA                PL     X4,TLD3     N.IT = MAX( N.IT , C+1 )                 CGIA        4711          I
          -CCG16
CGIA                SA6    A4                                                   CGIA        4712          I
          -CCG16
CGIA                                                                            CGIA        4713          I
          -CCG16
CGIA      TLD3      LX5    -SI.CAP                                              CGIA        4714          I
          -CCG16
CGIA                MX1    -SI.CAL                                              CGIA        4715          I
          -CCG16
CGIA                BX5    X1*X5                                                CGIA        4716          I
          -CCG16
CGIA                BX7    X5+X7       CA[PI] = C                               CGIA        4717          I
          -CCG16
CGIA                LX7    SI.CAP                                               CGIA        4718          I
          -CCG16
CGIA                S"SZ"  2           SZ = 2                                   CGIA        4719          I
          -CCG16
CGIA                SA7    A5                                                   CGIA        4720          I
          -CCG16
CGIA                EQ     WII1                                                 CGIA        4721          I
          -CCG16
CGIA      PLD       SPACE  2,12                                                 CGIA        4722          I
          -CCG16
CGIA                PROCESS (PLD,PST,S)                                         CGIA        4723          I
          -CCG16
CGIA                AX7    SI.CAP                                               CGIA        4724          I
          -CCG16
CGIA                ZR     X7,WII1     IF CA[PI] = 0                            CGIA        4725          I
          -CCG16
CGIA                S"SZ"  B1+B1       SZ = 2                                   CGIA        4726          I
          -CCG16
CGIA                EQ     WII1                                                 CGIA        4727          I
          -CCG16
CGIA      LD        SPACE  3,14                                                 CGIA        4728          I
          -CCG16
CGIA                PROCESS (LD,ST,STT,ILD)                                     CGIA        4729          I
          -FCC1842X -CCG16
CGIA                PROCESS ILD                                                 FCC1842X       2          I
          -CCG16
CGIA                SA2    =XS$IT                                               FCC1842X       3          I
          -CCG16
CGIA                LX5    -SI.IHP                                              FCC1842X       4          I
          -CCG16
CGIA                BX1    X2-X5                                                FCC1842X       5          I
          -CCG16
CGIA                SB4    X1                                                   FCC1842X       6          I
          -CCG16

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                LX5    SI.IHP                                          FCC1842X      7          I
          -CCG16
CGIA                ZR     X1,TLD      IF IH[PI] = IT.                     FCC1842X      8          I
          -CCG16
CGIA                                                                       FCC1842X      9          I
          -CCG16
CGIA                PROCESS (LD,ST,STT)                                    FCC1842X     10          I
          -CCG16
CGIA                AX7    SI.CAIHP                                        CGIA        4730          I
          -CCG16
CGIA                ZR     X7,WII1     IF CAIH[PI] = 0                     CGIA        4731          I
          -CCG16
CGIA                S"SZ"  B1+B1       SZ = 2                              CGIA        4732          I
          -CCG16
CGIA                                                                       CGIA        4733          I
          -CCG16
CGIA                MX1    -IH.IL                                          CGIA        4734          I
          -CCG16
CGIA                LX1    IH.IP                                           CGIA        4735          I
          -CCG16
CGIA                BX2    -X1*X7                                          CGIA        4736          I
          -CCG16
CGIA                NZ     X2,LD1      IF I[PI] " 0  */ NOT IN SYM         CGIA        4737          I
          -CCG16
CGIA                IF     -DEF,WB.MATP,1                                  CGIA        4738          I
          -CCG16
CGIA                IFNE   HC.FPAS,0,3                                     CGIA        4739          I
          -CCG16
CGIA                LX6    B1,X7                                           CGIA        4740          I
          -CCG16
CGIA                SB3    X7+B6                                           CGIA        4741          I
          -CCG16
CGIA                SA3    B3+X6       WORDB = SYM(3*IH+1)                 CGIA        4742          I
          -CCG16
CGIA                                                                       CGIA        4743          I
          -CCG16
CGIA       .MAT     IF     DEF,WB.MATP                                     CGIA        4744          I
          -CCG16
CGIA                SX1    B1                                              CGIA        4745          I
          -CCG16
CGIA                LX1    WB.MATP                                         CGIA        4746          I
          -CCG16
CGIA                BX6    X1+X3       MAT[SYM(3*IH+1)] = 1                CGIA        4747          I
          -CCG16
CGIA                SA6    A3                                              CGIA        4748          I
          -CCG16
CGIA       .MAT     ENDIF                                                  CGIA        4749          I
          -CCG16
CGIA                                                                       CGIA        4750          I
          -CCG16
CGIA       .FPAS    IFNE   HC.FPAS,0                                       CGIA        4751          I
          -CCG16

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                LX3    59-WB.FPP                                          CGIA        4752          I
          -CCG16
CGIA      +         PL     X3,*+1      IF ^FP[WORDB]                          CGIA        4753          I
          -CCG16
CGIA                RJ     CSR         COUNT SUB REF                          CGIA        4754          I
          -CCG16
CGIA       .FPAS    ENDIF                                                      CGIA        4755          I
          -CCG16
CGIA                                                                          CGIA        4756          I
          -CCG16
CGIA       LD1      LX5    59-SI.H2P                                          CGIA        4757          I
          -CCG16
CGIA                PL     X5,WII1     IF ^HI[PI]                             CGIA        4758          I
          -CCG16
CGIA                SA5    A5+1        PI = PI + 1   */ SKIP WORD WITH *H2* IN IT  CGIA   4759          I
          -CCG16
CGIA                EQ     WII1                                               CGIA        4760          I
          -CCG16
CGIA       LDC      SPACE  2                                                  CGIA        4761          I
          -CCG16
CGIA                PROCESS LDC                                               CGIA        4762          I
          -CCG16
CGIA                LX5    -SI.CAP                                            CGIA        4763          I
          -CCG16
CGIA                SX6    B1                                                 CGIA        4764          I
          -CCG16
CGIA                S"SZ"  B1+B1       SZ = 2                                 CGIA        4765          I
          -CCG16
CGIA                SB3    A0                                                 CGIA        4766          I
          -CCG16
CGIA                SA6    B3+X5       CUT(CA[SI]) = 1                        CGIA        4767          I
          -CCG16
CGIA                EQ     WII1                                               CGIA        4768          I
          -CCG16
CGIA       JPX      SPACE  3                                                  CGIA        4769          I
          -CCG16
CGIA      *         JUMPS                                                     CGIA        4770          I
          -CCG16
CGIA                                                                          CGIA        4771          I
          -CCG16
CGIA                PROCESS (JPX,JPBB)                                        CGIA        4772          I
          -CCG16
CGIA                SB4    SI.IHP                                             CGIA        4773          I
          -CCG16
CGIA                RJ     RLV         CHECK REFERENCED LABEL                 CGIA        4774          I
          -CCG16
CGIA                S"SZ"  2           SZ = 2                                 CGIA        4775          I
          -CCG16
CGIA                EQ     WII1                                               CGIA        4776          I
          -CCG16
CGIA                                                                          CGIA        4777          I
          -CCG16

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                PROCESS JIN                                                    CGIA        4778          I
          -CCG16
CGIA                SB4     SI.IHP                                                 CGIA        4779          I
          -CCG16
CGIA                RJ      RLV         CHECK REFERENCED LABEL                     CGIA        4780          I
          -CCG16
CGIA                EQ      UJP1                                                   CGIA        4781          I
          -CCG16
CGIA                                                                               CGIA        4782          I
          -CCG16
CGIA                PROCESS UJP                                                    CGIA        4783          I
          -CCG16
CGIA                SB4     B0                                                     CGIA        4784          I
          -CCG16
CGIA                RJ      RLV         CHECK REFERENCED LABEL                     CGIA        4785          I
          -CCG16
CGIA                                                                               CGIA        4786          I
          -CCG16
CGIA      *         UNCONDITIONAL JUMP, FORCE UPPER AFTER                          CGIA        4787          I
          -CCG16
CGIA                                                                               CGIA        4788          I
          -CCG16
CGIA                PROCESS RJ3                                                    CGIA        4789          I
          -CCG16
CGIA       UJP1     SX6     "PC"+B1     S = PC + 1                                 CGIA        4790          I
          -CCG16
CGIA                S"BL"   "BL"+B1     BLEN = BLEN + 1                            CGIA        4791          I
          -CCG16
CGIA                S"PC"   -4          PC = 0                                     CGIA        4792          I
          -CCG16
CGIA                SA5     A5+B1       PI = PI + 1                                CGIA        4793          I
          -CCG16
CGIA                MI      X6,WII2     IF S < 3      */ OLD(PC)+1 < 3             CGIA        4794          I
          -CCG16
CGIA                S"BL"   "BL"+B1     BLEN = BLEN + 1                            CGIA        4795          I
          -CCG16
CGIA                EQ      WII2                                                   CGIA        4796          I
          -CCG16
CGIA       RJXJ     SPACE   2,8                                                    CGIA        4797          I
          -CCG16
CGIA       .RJXJ    IFNE    HC.RJXJ,0                                              CGIA        4798          I
          -CCG16
CGIA                                                                               CGIA        4799          I
          -CCG16
CGIA                PROCESS RJXJ                                                   CGIA        4800          I
          -CCG16
CGIA                SX6     "PC"+4                                                 CGIA        4801          I
          -CCG16
CGIA                ZR      X6,RJXJ1    IF PC = 0                                  CGIA        4802          I
          -CCG16
CGIA                S"PC"   -4          PC = 0        */ FORCE UPPER BEFORE        CGIA        4803          I
          -CCG16

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                S"BL"   "BL"+1      BLEN = BLEN + 1                                CGIA        4804          I
            -CCG16
CGIA         RJXJ1  SA5     A5+B1       PI = PI + 1                                    CGIA        4805          I
            -CCG16
CGIA                S"BL"   "BL"+B1     BLEN = BLEN + 1                                CGIA        4806          I
            -CCG16
CGIA                EQ      WII2                                                       CGIA        4807          I
            -CCG16
CGIA                                                                                   CGIA        4808          I
            -CCG16
CGIA         .RJXJ  ENDIF                                                              CGIA        4809          I
            -CCG16
CGIA         RJ6    SPACE   3,14                                                       CGIA        4810          I
            -CCG16
CGIA         .RJ6   IFNE    HC.RJ6,0                                                   CGIA        4811          I
            -CCG16
CGIA                                                                                   CGIA        4812          I
            -CCG16
CGIA                PROCESS RJ6                                                        CGIA        4813          I
            -CCG16
CGIA                SX6     "PC"+4                                                     CGIA        4814          I
            -CCG16
CGIA                ZR      X6,RJ61     IF PC = 0                                      CGIA        4815          I
            -CCG16
CGIA                S"PC"   -4          PC = 0                                         CGIA        4816          I
            -CCG16
CGIA                S"BL"   "BL"+1      BLEN = BLEN + 1                                CGIA        4817          I
            -CCG16
CGIA         RJ61   BSS     0                                                          CGIA        4818          I
            -CCG16
CGIA         .FPAS  IFNE    HC.FPAS,0                                                  CGIA        4819          I
            -CCG16
CGIA                SB3     X5+B6                                                      CGIA        4820          I
            -CCG16
CGIA                LX6     B1,X5                                                      CGIA        4821          I
            -CCG16
CGIA                SA3     B3+X6                                                      CGIA        4822          I
            -CCG16
CGIA                LX3     59-WB.FPP                                                  CGIA        4823          I
            -CCG16
CGIA        +       PL      X3,*+1      IF ^FP[WORDB]                                  CGIA        4824          I
            -CCG16
CGIA                RJ      CSR         COUNT SUB REF                                  CGIA        4825          I
            -CCG16
CGIA         .FPAS  ENDIF                                                              CGIA        4826          I
            -CCG16
CGIA                                                                                   CGIA        4827          I
            -CCG16
CGIA                LX5     -SI.CAP                                                    CGIA        4828          I
            -CCG16
CGIA                SX1     X5-7777B                                                   CGIA        4829          I
            -CCG16

            0         1         2         3         4         5         6         7         8
            12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                MI     X1,RJ62     IF CA[R1] < 7777B                          CGIA        4830          I
          -CCG16
CGIA                S"BL"  "BL"+1      BLEN = BLEN + 1                            CGIA        4831          I
          -CCG16
CGIA       RJ62     SA5    A5+B1       PI = PI + 1                                CGIA        4832          I
          -CCG16
CGIA                S"BL"  "BL"+B1     BLEN = BLEN + 1  */ COUNT RJ WORD          CGIA        4833          I
          -CCG16
CGIA                EQ     WII2                                                   CGIA        4834          I
          -CCG16
CGIA                                                                              CGIA        4835          I
          -CCG16
CGIA       .RJ6     ENDIF                                                          CGIA        4836          I
          -CCG16
CGIA       CSR      SPACE  3,14                                                   CGIA        4837          I
          -CCG16
CGIA       .FPAS    IFNE   HC.FPAS,0                                              CGIA        4838          I
          -CCG16
CGIA      **        CSR - COUNT *SUB* REFERENCE                                   CGIA        4839          I
          -CCG16
CGIA      *                                                                       CGIA        4840          I
          -CCG16
CGIA      *         ENTRY  (A3) = ADDRESS OF WORD B                               CGIA        4841          I
          -CCG16
CGIA      *                (X3) = SHIFT(WORDB,59-WB.FPP)                          CGIA        4842          I
          -CCG16
CGIA      *                (X5) = [PI]                                            CGIA        4843          I
          -CCG16
CGIA                                                                              CGIA        4844          I
          -CCG16
CGIA       CSR      ROUTINE                                                       CGIA        4845          I
          -CCG16
CGIA                SA1    O.FPT                                                  CGIA        4846          I
          -CCG16
CGIA                LX3    1+WB.FPP-WB.FPOP                                       CGIA        4847          I
          -CCG16
CGIA                MX4    -WB.FPOL                                               CGIA        4848          I
          -CCG16
CGIA                BX2    -X4*X3                                                 CGIA        4849          I
          -CCG16
CGIA                SB3    X2                                                     CGIA        4850          I
          -CCG16
CGIA                SA2    B3+X1                                                  CGIA        4851          I
          -CCG16
CGIA                SX4    B1                                                     CGIA        4852          I
          -CCG16
CGIA                IX6    X2+X4       FPIT(FPO) = FPIT(FPO) + 1                  CGIA        4853          I
          -CCG16
CGIA                SA6    A2                                                     CGIA        4854          I
          -CCG16
CGIA                SA1    LSR         OLSR = LSR                                 CGIA        4855          I
          -CCG16

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                SX7    "BL"                                                    CGIA        4856          I
          -CCG16
CGIA                PX7    B3,X7                                                   CGIA        4857          I
          -CCG16
CGIA                SA7    A1          LSR = PACK(FPO,BLEN)                        CGIA        4858          I
          -CCG16
CGIA                BX2    X1-X7                                                   CGIA        4859          I
          -CCG16
CGIA                NZ     X2,CSR      IF OLSR " LSR */ NOT REF TO SAME FP IN A WD CGIA        4860          I
          -CCG16
CGIA                                                                               CGIA        4861          I
          -CCG16
CGIA                IX6    X6+X4       FPIT(FPO) = FPIT(FPO) + 1 */ COUNT *DELAY*  CGIA        4862          I
          -CCG16
CGIA                SA6    A6                                                      CGIA        4863          I
          -CCG16
CGIA                EQ     CSR                                                     CGIA        4864          I
          -CCG16
CGIA       .FPAS    ENDIF                                                          CGIA        4865          I
          -CCG16
CGIA       LAB      SPACE  3,14                                                    CGIA        4866          I
          -CCG16
CGIA      **        LAB    CA,IH - DEFINE LABEL ADDRESS                            CGIA        4867          I
          -CCG16
CGIA                                                                               CGIA        4868          I
          -CCG16
CGIA                PROCESS ENT                                                    CGIA        4869          I
          -CCG16
CGIA                SX4    B1                                                      CGIA        4870          I
          -CCG16
CGIA                LX4    SI.CAP                                                  CGIA        4871          I
          -CCG16
CGIA                BX5    X4+X5       CA = 1                                      CGIA        4872          I
          -CCG16
CGIA                                                                               CGIA        4873          I
          -CCG16
CGIA                PROCESS LAB                                                    CGIA        4874          I
          -CCG16
CGIA                SX6    "PC"+4                                                  CGIA        4875          I
          -CCG16
CGIA                ZR     X6,LAB1     IF PC = 0                                   CGIA        4876          I
          -CCG16
CGIA                S"PC"  -4          PC = 0                                      CGIA        4877          I
          -CCG16
CGIA                S"BL"  "BL"+1      BLEN = BLEN + 1                             CGIA        4878          I
          -CCG16
CGIA                                                                               CGIA        4879          I
          -CCG16
CGIA       LAB1     SA1    CC$LBO                                                  CGIA        4880          I
          -CCG16
CGIA                SX2    B1                                                      CGIA        4881          I
          -CCG16

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                LX2    WC.RLP                                          CGIA        4882          I
          -CCG16
CGIA                SX3    "BL"                                            CGIA        4883          I
          -CCG16
CGIA                LX1    WC.RBP                                          CGIA        4884          I
          -CCG16
CGIA                BX2    X1+X2                                           CGIA        4885          I
          -CCG16
CGIA                BX6    X2+X3       ADW = WC(1,0,CG.LBO,BLEN)           CGIA        4886          I
          -CCG16
CGIA                AX7    IH.IP                                           CGIA        4887          I
          -CCG16
CGIA                MX1    -IH.IL                                          CGIA        4888          I
          -CCG16
CGIA                BX7    -X1*X7                                          CGIA        4889          I
          -CCG16
CGIA                ZR     X7,LAB2     IF I[PI] = 0                        CGIA        4890          I
          -CCG16
CGIA                                                                       CGIA        4891          I
          -CCG16
CGIA                SA3    O.GLT-1+X7  BASE                                CGIA        4892          I
          -CCG16
CGIA                MX4    -IH.HL                                          CGIA        4893          I
          -CCG16
CGIA                BX7    -X4*X5                                          CGIA        4894          I
          -CCG16
CGIA                SB3    X7                                              CGIA        4895          I
          -CCG16
CGIA                SA6    X3+B3       TBL(H) = ADW                        CGIA        4896          I
          -CCG16
CGIA                EQ     LAB3                                            CGIA        4897          I
          -CCG16
CGIA                                                                       CGIA        4898          I
          -CCG16
CGIA      LAB2      LX7    B1,X5                                           CGIA        4899          I
          -CCG16
CGIA                SB3    B6+X5                                           CGIA        4900          I
          -CCG16
CGIA                SX7    X7+B1                                           CGIA        4901          I
          -CCG16
CGIA                SA6    B3+X7       SYM(3*IH+2) = ADW                   CGIA        4902          I
          -CCG16
CGIA                                                                       CGIA        4903          I
          -CCG16
CGIA      LAB3      LX5    -SI.CAP                                         CGIA        4904          I
          -CCG16
CGIA                S"BL"  "BL"+X5     BLEN = BLEN + CA[PI]                CGIA        4905          I
          -CCG16
CGIA                SA5    A5+1        PI = PI + 1                         CGIA        4906          I
          -CCG16
CGIA                EQ     WII2                                            CGIA        4907          I
          -CCG16

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA      RLV      SPACE  3,14                                                    CGIA        4908          I
          -CCG16
CGIA      **       RLV - CHECK REFERENCED LABEL VALUE FOR SUBSTITUTION             CGIA        4909          I
          -CCG16
CGIA      *                                                                        CGIA        4910          I
          -CCG16
CGIA      *        ENTRY  (B4) = SHIFT COUNT TO FIND *IH* IN *PI* WORD             CGIA        4911          I
          -CCG16
CGIA                                                                               CGIA        4912          I
          -CCG16
CGIA      RLV      ROUTINE                                                         CGIA        4913          I
          -CCG16
CGIA               AX6    B4,X5                                                    CGIA        4914          I
          -CCG16
CGIA               MX3    -IH.IL                                                   CGIA        4915          I
          -CCG16
CGIA               SB3    B6+X6                                                    CGIA        4916          I
          -CCG16
CGIA               LX3    IH.IP                                                    CGIA        4917          I
          -CCG16
CGIA               BX7    -X3*X6                                                   CGIA        4918          I
          -CCG16
CGIA               NZ     X7,RLV      IF I[PI] " 0  */ NOT IN *SYM*                CGIA        4919          I
          -CCG16
CGIA               SX6    X6                                                       CGIA        4920          I
          -CCG16
CGIA               LX7    B1,X6                                                    CGIA        4921          I
          -CCG16
CGIA               SA3    B3+X7       WORDB = SYM(3*H+1)                           CGIA        4922          I
          -CCG16
CGIA               LX3    59-WB.LCP                                                CGIA        4923          I
          -CCG16
CGIA               PL     X3,RLV      IF ^LC[WORDB]                                CGIA        4924          I
          -CCG16
CGIA                                                                               CGIA        4925          I
          -CCG16
CGIA               SA2    A3-B1       WORDA = WORDB - 1                            CGIA        4926          I
          -CCG16
CGIA               MX1    -WA.LCOL                                                 CGIA        4927          I
          -CCG16
CGIA               LX2    -WA.LCOP                                                 CGIA        4928          I
          -CCG16
CGIA               BX3    -X1*X2      L = LCO[WORDA]                               CGIA        4929          I
          -CCG16
CGIA               SA1    =XO$LCT                                                  CGIA        4930          I
          -CCG16
CGIA               SA2    =XO$BIT                                                  CGIA        4931          I
          -CCG16
CGIA               SB2    X1                                                       CGIA        4932          I
          -CCG16
CGIA               SB3    X2                                                       CGIA        4933          I
          -CCG16

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                                                                                    CGIA        4934          I
          -CCG16
CGIA       RLV1     SA1    B2+X3       LCW = LCT(L)                                     CGIA        4935          I
          -CCG16
CGIA               SA3    =XBN#                                                         CGIA        4936          I
          -CCG16
CGIA               MX5    -BI.OLNL                                                      CGIA        4937          I
          -CCG16
CGIA                                                                                    CGIA        4938          I
          -CCG16
CGIA       RLV2     BX7    X1-X3                                                         CGIA        4939          I
          -CCG16
CGIA               BX7    -X5*X7                                                        CGIA        4940          I
          -CCG16
CGIA               ZR     X7,RLV3     IF K = HBN[LCW]                                   CGIA        4941          I
          -CCG16
CGIA               SA3    B3+X3                                                         CGIA        4942          I
          -CCG16
CGIA               LX3    -BI.OLNP                                                      CGIA        4943          I
          -CCG16
CGIA               BX3    -X5*X3      K = OLN[BIT(K)]                                   CGIA        4944          I
          -CCG16
CGIA               NZ     X3,RLV2     IF K " 0                                          CGIA        4945          I
          -CCG16
CGIA                                                                                    CGIA        4946          I
          -CCG16
CGIA               LX1    -LC.NEXTP                                                     CGIA        4947          I
          -CCG16
CGIA               SX3    X1          L = NEXT[LCW]                                     CGIA        4948          I
          -CCG16
CGIA               NZ     X3,RLV1     IF L " 0      */ NOT END OF CHAIN                 CGIA        4949          I
          -CCG16
CGIA               EQ     RLV                                                           CGIA        4950          I
          -CCG16
CGIA                                                                                    CGIA        4951          I
          -CCG16
CGIA       RLV3     LX1    -LC.GLNP                                                      CGIA        4952          I
          -CCG16
CGIA               SX7    X1+I.GL     IH = GLN[LCW] + I.GL                              CGIA        4953          I
          -CCG16
CGIA               LX6    B4                                                            CGIA        4954          I
          -CCG16
CGIA               BX5    X5-X6                                                         CGIA        4955          I
          -CCG16
CGIA               LX7    B4                                                            CGIA        4956          I
          -CCG16
CGIA               BX7    X5+X7       SUBSTITUTE NEW VALUE                              CGIA        4957          I
          -CCG16
CGIA               SA7    A5                                                            CGIA        4958          I
          -CCG16
CGIA               EQ     RLV                                                           CGIA        4959          I
          -CCG16

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

```
CGIA                                                                                    CGIA        4960          I
          -CCG16
CGIA                QUAL   *                                                           CGIA        4961          I
          -CCG16
CGIA       .Z       SET    *-L.WII     AVAILABLE SPACE                                 CGIA        4962          I
          -CCG16
CGIA                IFLT   .Z,0,1                                                      CGIA        4963          I
          -CCG16
CGIA                BSS    -.Z                                                         CGIA        4964          I
          -CCG16
CGIA                USE    *                                                           CGIA        4965          I
          -CCG16
CGIA                                                                                   CGIA        4966          I
          -CCG16
CGIA                ECHO   2,Z=(CG$RBT,CG$CUB,FSU#,CG$DSA,CG$EP,WII#)                  CGIA        4967          I
          -CCG16
CGIA       Z        EQU    /WII/Z                                                      CGIA        4968          I
          -CCG16
CGIA                ENTRY  Z                                                           CGIA        4969          I
          -CCG16
CGIA                                                                                   CGIA        4970          I
          -CCG16
CGIA                                                                                   CGIA        4971          I
          -CCG16
CGIA       .TS      ENDIF                                                               CGIA        4972          I
          -CCG16
CGIA       OC.JT    TITLE  OC.JT - PSEUDO OPERATION AND MACHINE INST JUMP TABLE         CGIA        4973     A
CGIA      **        DEFINE JUMP TABLE FOR PSEUDO OP"S, MACHINE INSTRUCTIONS             CGIA        4974     A
CGIA      *         AND PSEUDO"S USED BY THE ASSEMBLER                                  CGIA        4975     A
CGIA                                                                                   CGIA        4976     A
CGIA       POL      MICRO  1,,                                                          CGIA        4977     A
CGIA                                                                                   CGIA        4978     A
CGIA                MACRO  POD,NAM                                                     CGIA        4979     A
CGIA                IFC    NE,/"POL"//,2                                               CGIA        4980     A
CGIA       POL      MICRO  1,,/NAM,"POL"/                                              CGIA        4981     A
CGIA                SKIP   1                                                           CGIA        4982     A
CGIA       POL      MICRO  1,,/NAM/                                                    CGIA        4983     A
CGIA                ENDM                                                               CGIA        4984     A
CGIA                                                                                   CGIA        4985     A
CGIA       MAX.OC   SET    OC.LDC+1                                                    CGIA        4986     A
CGIA                IFNE   HC.TS,0,1                                                   CGIA        4987     A
CGIA       MAX.OC   EQU    OC.RS                                                       CGIA        4988     A
CGIA                                                                                   CGIA        4989     A
CGIA                MACRO  OPR,N                                                       CGIA        4990     A
CGIA                IFLT   OC.N,MAX.OC                                                 CGIA        4991     A
CGIA                IF     DEF,)N,2                                                    CGIA        4992     A
CGIA                VFD    24/)N                                                       CGIA        4993     A
CGIA                SKIP   1                                                           CGIA        4994     A
CGIA                VFD    24/0                                                        CGIA        4995     A
CGIA                IF     DEF,/WII/.N,2                                               CGIA        4996          I
          -CCG16
CGIA                VFD    18//WII/.N-/WII/WII1                                        CGIA        4997          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGIA

                -CCG16
      CGIA                SKIP   1                                                CGIA        4998          I
                -CCG16
      CGIA                VFD    18/0        N                                    CGIA        4999          I
                -CCG16
      CGIA                VFD    18/0                                             CCG16         63     A
      CGIA                IF     DEF,.N,2                                         CGIA        5000     A
      CGIA                VFD    18/.N                                            CGIA        5001     A
      CGIA                SKIP   1                                                CGIA        5002     A
      CGIA                VFD    18/*+1S17         N                              CGIA        5003     A
      CGIA                ENDIF                                                   CGIA        5004     A
      CGIA                ENDM                                                    CGIA        5005     A
      CGIA                                                                        CGIA        5006     A
      CGIA                LIST   G                                                CGIA        5007     A
      CGIA                HERE                                                    CGIA        5008     A
      CGIA                                                                        CGIA        5009     A
      CGIA     *CALL      PSODEFS                                                 CGIA        5010     A
      CGIA                                                                        CGIA        5011     A
      CGIA                ECHO   4,Z=("POL")                                      CGIA        5012     A
      CGIA                IF     DEF,.Z,2                                         CGIA        5013     A
      CGIA                VFD    42/,18/.Z                                        CGIA        5014     A
      CGIA                SKIP   1                                                CGIA        5015     A
      CGIA                VFD    42/,18/*+1S17     Z                              CGIA        5016     A
      CGIA                                                                        CGIA        5017     A
      CGIA      OC.AT     BSS    0                                                CGIA        5018     A
      CGIA     *CALL      OPRDEFS                                                 CGIA        5019     A
      CGIA                                                                        CGIA        5020     A
      CGIA                END                                                     CGIA        5021     A

           SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - CGIA

                IDENTIFIER                TOTAL    ACTIVE

                CGIA                       5021      4232
                CCG14                         6         6
                FCC1842X                     10         0
                CCG16                        51        47
                CCG18                         7         7
                CCGA002                       8         8
                CCGA005                      40        38
                CCGA007                      24        24
                CCG29                         3         1
                CCGA008                       2         2
                CCG0001                       1         1
                CCGA067                      35        35
                CCGA068                      19        19
                CCGA072                      20        20

                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MACROS

      MACROS    *DECK      MACROS                                                 MACROS         1     A
      MACROS               IDENT  MACRO_S                                        MACROS         2     A

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MACROS

```
MACROS    MACROS   TITLE  MACRO_S - HOST COMPILER STORAGE MACRO SKELTONS.          MACROS      3    A
MACROS   *CALL     SSTCALL                                                         MACROS      4    A
MACROS             SPACE  3                                                        MACROS      5    A
MACROS   **        MACRO_S - HOST COMPILER STORAGE MACRO SKELTONS.                 MACROS      6    A
MACROS   *                                                                         MACROS      7    A
MACROS   *         *MACRO_S* CONTAINS THE DEFINITIONS OF THE HOST COMPILERS        MACROS      8    A
MACROS   *         MACROS THAT GENERATE STORAGE RESERVATION AND INITIALIZATION     MACROS      9    A
MACROS   *         DIRECTIVES IN THE RELOCATABLE BINARY.  FOR EACH MACRO IN        MACROS     10    A
MACROS   *         THE HOST COMPILERS ASSEMBLY TEXT ( FTNMAC,PLIMAC,ETC. ) THERE   MACROS     11    A
MACROS   *         IS A CORRESPONDING *SMACRO* DEFINITION IN A COMDECK *SMACROS*   MACROS     12    A
MACROS   *         WHICH RESIDES ON THE IT"S OLDPL.  THE RULES FOR CREATING THIS   MACROS     13    A
MACROS   *         DECK AND THE SMACROS ARE DISCUSSED BELOW.                       MACROS     14    A
MACROS             SPACE  3                                                        MACROS     15    A
MACROS    N.MMP    EQU    6           MAX MACRO PARAMS                             MACROS     16    A
MACROS                                                                             MACROS     17    A
MACROS   *         DEFINE ORDERING OF LOCAL BLOCKS                                 MACROS     18    A
MACROS                                                                             MACROS     19    A
MACROS             USE    PARAMB      PARAMETER BUFFERS                            MACROS     20    A
MACROS             USE    INDEXT      MACRO OPCODE INDEX TABLE                     MACROS     21    A
MACROS             ENTRY  F$MACS                                                   MACROS     22    A
MACROS    F$MACS   BSS    0                                                        MACROS     23    A
MACROS             USE    SKELS       MACRO SKELTONS                               MACROS     24    A
MACROS                                                                             MACROS     25    A
MACROS             USE    PARAMB                                                   MACROS     26    A
MACROS    PBUF     ENTRY. 0,#         0 WORD                                       MACROS     27    A
MACROS             BSS    N.MMP       MACRO PARAMETER WORDS                        MACROS     28    A
MACROS    L0       BSSZ   N.MMP+1     MACRO PARAMTER WORD                          MACROS     29    A
MACROS    Z        SET    L0-PBUF                                                 MACROS     30    A
MACROS    SS#BIAS  EQENT  377777B-Z   - INDEX FROM PBUF TO LOCAL SYMS              MACROS     31    A
MACROS             ECHO   1,Z=(1,2,3,4,5,6)                                        MACROS     32    A
MACROS    L_Z      EQU    L0+Z                                                    MACROS     33    A
MACROS                                                                             MACROS     34    A
MACROS             ENTRY  F$MIC       MICRO STRING VALUES - 60/0L_STRNG,60/6*NCHR  MACROS     35    A
MACROS    F$MIC    BSS    0                                                        MACROS     36    A
MACROS             ECHO   1,I=(1,2,3,4)                                            MACROS     37    A
MACROS    M_I      BSS    2                                                        MACROS     38    A
MACROS    ORG#CTR  ENTRY. 0                                                        MACROS     39    A
MACROS                                                                             MACROS     40    A
MACROS             ECHO   ,Z=(1,2,3,4,5,6)                                         MACROS     41    A
MACROS             QUAL   CON                                                      MACROS     42    A
MACROS    C_Z      =      0                                                        MACROS     43    A
MACROS             QUAL   SYM                                                      MACROS     44    A
MACROS    S_Z      =      0                                                        MACROS     45    A
MACROS             ENDD                                                            MACROS     46    A
MACROS             QUAL                                                            MACROS     47    A
MACROS    IF.XX    SPACE  2,7                                                      MACROS     48    A
MACROS   *         DEFINE IF.XX VALUES                                             MACROS     49    A
MACROS                                                                             MACROS     50    A
MACROS    .Z       SET    0                                                        MACROS     51    A
MACROS             ECHO   2,XX=("RELOPS")                                          MACROS     52    A
MACROS    IF.XX    EQU    .Z                                                       MACROS     53    A
MACROS    .Z       SET    .Z+1                                                     MACROS     54    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MACROS

```
MACROS                                                                  MACROS     55    A
MACROS   *         DEFINE ORDINALS OF FIXED LOCAL BLOCKS FOR USE. MACRO, ETC.   MACROS     56    A
MACROS                                                                  MACROS     57    A
MACROS             IFC    NE,/"HC.FLB"//,4                                      MACROS     58    A
MACROS    .Z       SET    1S15+0                                                MACROS     59    A
MACROS             ECHO   2,X=("HC.FLB")                                        MACROS     60    A
MACROS    X$       EQU    .Z                                                    MACROS     61    A
MACROS    .Z       SET    .Z+1                                                  MACROS     62    A
MACROS                                                                  MACROS     63    A
MACROS   *         DEFINE OPERATOR VALUE MICROS FOR SET. MACRO                  MACROS     64    A
MACROS                                                                  MACROS     65    A
MACROS    +        MICRO  1,,/0/                                                MACROS     66    A
MACROS    -        MICRO  1,,/1/                                                MACROS     67    A
MACROS    *        MICRO  1,,/2/                                                MACROS     68    A
MACROS    /        MICRO  1,,/3/                                                MACROS     69    A
MACROS    LITORD   SPACE  2,14                                                  MACROS     70    A
MACROS   **        LITORD? - DEFINE LITERAL ORDINAL IN *PARAMB* BLOCK           MACROS     71    A
MACROS   *                                                              MACROS     72    A
MACROS   * ORD     LITORD? CONSTANT                                             MACROS     73    A
MACROS                                                                  MACROS     74    A
MACROS             MACRO  LITORD?,L,VAL                                         MACROS     75    A
MACROS             IF     -DEF,/PARAMB/.VAL,6                                   MACROS     76    A
MACROS             USE    PARAMB                                                MACROS     77    A
MACROS             QUAL   PARAMB                                                MACROS     78    A
MACROS    .VAL     =      *-PBUF                                                MACROS     79    A
MACROS             QUAL   *                                                     MACROS     80    A
MACROS             DATA   VAL                                                   MACROS     81    A
MACROS             USE    *                                                     MACROS     82    A
MACROS   *                                                              MACROS     83    A
MACROS    L        SET    /PARAMB/.VAL                                          MACROS     84    A
MACROS             ENDM                                                         MACROS     85    A
MACROS                                                                  MACROS     86    A
MACROS             QUAL   PARAMB                                                MACROS     87    A
MACROS    .0       EQU    0                                                     MACROS     88    A
MACROS             QUAL                                                         MACROS     89    A
MACROS    MAC-DEF  TITLE  MACRO DEFINITIONS                                     MACROS     90    A
MACROS   **        SMACRO - DEFINE THE BEGINNING OF A STORAGE MACRO DEFINITION. MACROS     91    A
MACROS   *                                                              MACROS     92    A
MACROS   * NAME    SMACRO PLIST,LAB                                             MACROS     93    A
MACROS   *         NAME - MACRO NAME ( @ 6 CHARACTERS )                         MACROS     94    A
MACROS   *         PLIST - A PARENTHESIZED LIST OF THE ARGUMENT NAMES ( @ 6 ARGS) MACROS     95    A
MACROS   *         LAB  - = LAB IF THE FIRST ARG IS TO BE PLACED IN THE LABEL   MACROS     96    A
MACROS   *               FIELD WHEN THE MACRO IS LISTED.                        MACROS     97    A
MACROS   *                                                              MACROS     98    A
MACROS   *         THE ARE NAME LIST IS OF THE FORM - (PI=ANAME,...) OR (PI,... ) MACROS     99    A
MACROS   *         WHERE 1 @ I @ 6 , AND P IS THE ARGUMENT TYPE  -              MACROS    100    A
MACROS   *         C - 18 BIT CONSTANT, S - SYMTAB ORDINAL, Q - QUAL BLOCK TABLE MACROS    101    A
MACROS   *         ORDINAL, B - BLOCK TABLE ORDINAL, AND M - MICRO TABLE ORDINAL. MACROS    102    A
MACROS   *         LOCAL SYMBOL ( GENERATED WITHIN ) THE MACRO ARE *LI*, WHERE  MACROS    103    A
MACROS   *         1 @ I @ 6 .  THEY ARE NOT LISTED ON THE *SMACRO* LINE.       MACROS    104    A
MACROS   *         *ANAME* IS ANY SYMBOLIC NAME AND "ANAME" MAY BE REFERENCED IN MACROS    105    A
MACROS   *         THE MACRO DEFINITION WHERE EVER THE CORRESPONDING PARAMETER  MACROS    106    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MACROS

```
MACROS   *         MAY APPEAR.                                                  MACROS      107    A
MACROS   *                                                                      MACROS      108    A
MACROS   *         EXAMPLE -                                                    MACROS      109    A
MACROS   * PENTRY  SMACRO (S1=NAME,S2=ADDR,C1=VAL,C2=ER)                        MACROS      110    A
MACROS   *         THE CORRESPONDING *COMPASS* STATEMENT WOULD BE -             MACROS      111    A
MACROS   * PENTRY  MACRO  NAME,ADDR,VAL,ER                                      MACROS      112    A
MACROS   *         THE ENTRY POINT *M$NAME* IS DEFINED AS THE OPCODE VALUE.     MACROS      113    A
MACROS                                                                          MACROS      114    A
MACROS    .        EQU    0                                                     MACROS      115    A
MACROS    .LAB     EQU    1                                                     MACROS      116    A
MACROS                                                                          MACROS      117    A
MACROS             ECHO   1,Z=(C,B,S,Q,M,E,L),V=(1,2,3,4,5,6,7)                 MACROS      118    A
MACROS    AT.Z = V                                                              MACROS      119    A
MACROS                                                                          MACROS      120    A
MACROS             MACRO  SMACRO,NAME,PL,LAB                                    MACROS      121    A
MACROS             ERRNZ  .FWA        MISSING ENDS DIRECTIVE                    MACROS      122    A
MACROS    MN       MICRO  6,1, NAME                                             MACROS      123    A
MACROS             IFC    EQ,/"MN"//,1                                          MACROS      124    A
MACROS             ENTRY  M$NAME                                                MACROS      125    A
MACROS    M$NAME   =      .MN                                                   MACROS      126    A
MACROS    .MN      SET    .MN+1                                                 MACROS      127    A
MACROS             USE    SKELS                                                 MACROS      128    A
MACROS   +         VFD    36/6L;A                                               MACROS      129    A
MACROS             VFD    3/1-.LAB                                              MACROS      130    A
MACROS   *         SETUP A LIST OF ARG TYPES FOR O-LIST, PARAM ADDRESS"S AND EQV"  MACROS     131    A
MACROS    .N       SET    0                                                     MACROS      132    A
MACROS             IRP    PL                                                    MACROS      133    A
MACROS    .N       SET    .N+1                                                  MACROS      134    A
MACROS    P        MICRO  1,1,/PL/                                              MACROS      135    A
MACROS    PN       MICRO  1,2,/PL/                                              MACROS      136    A
MACROS             IFC    EQ,/"P"/B/,1                                          MACROS      137    A
MACROS    PN       MICRO  1,,/"PN"$/                                            MACROS      138    A
MACROS    EQV      MICRO  4,,/PL/                                               MACROS      139    A
MACROS             IFC    NE,/"EQV"//,1                                         MACROS      140    A
MACROS    "EQV"    MICRO  1,,/"PN"/                                             MACROS      141    A
MACROS             VFD    3/AT."P"                                              MACROS      142    A
MACROS             IFC    NE,/"P"/M/,1                                          MACROS      143    A
MACROS    "PN"     SET    PBUF+.N                                               MACROS      144    A
MACROS             IRP                                                          MACROS      145    A
MACROS             ERRPL  .N-1-N.MMP  ;A HAS MORE THAN N.MMP PARAMETERS         MACROS      146    A
MACROS             VFD    *P/.N                                                 MACROS      147    A
MACROS   *         FWA OF MACRO BODY, N.PARAM RELOC, N.LOCAL RELOC              MACROS      148    A
MACROS    .FWA     SET    *                                                     MACROS      149    A
MACROS    L.PR     SET    .FWA                                                  MACROS      150    A
MACROS    N.PR     SET    0                                                     MACROS      151    A
MACROS    .IFB     SET    0                                                     MACROS      152    A
MACROS             ENDM                                                         MACROS      153    A
MACROS                                                                          MACROS      154    A
MACROS             NOREF  .FWA,.LEN,N.PR,L.PR,.Z,Z,.N,.L                        MACROS      155    A
MACROS    .FWA     SET    0                                                     MACROS      156    A
MACROS    .MN      SET    128        BASE VALUE OF SMACRO OPCODES               MACROS      157    A
MACROS    ENDS     SPACE  2,14                                                  MACROS      158    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MACROS

```
MACROS   **        ENDS - TERMINATE A *SMACRO* DEFINITION                MACROS   159   A
MACROS   *                                                               MACROS   160   A
MACROS   *         ENDS                                                  MACROS   161   A
MACROS   *         ENDS CORRESPONDS TO THE COMPASS *ENDM* PSEUDO.        MACROS   162   A
MACROS                                                                   MACROS   163   A
MACROS    ENDS     MACRO                                                 MACROS   164   A
MACROS    .LEN     SET    *-.FWA                                         MACROS   165   A
MACROS    MAX.LM   MAX    .LEN,MAX.LM                                    MACROS   166   A
MACROS             IFC    NE,/"IFSTK"//,2                                MACROS   167   A
MACROS    IFSTK    MICRO  1,,                                            MACROS   168   A
MACROS             ERR    IF/ENDIF NESTING ERROR                         MACROS   169   A
MACROS   *                         EXPAND RELOC BYTES                    MACROS   170   A
MACROS             ECHO   6,T=(PR)                                       MACROS   171   A
MACROS             IFNE   N.T,0,5                                        MACROS   172   A
MACROS    Z        SET    1                                              MACROS   173   A
MACROS             DUP    N.T,3                                          MACROS   174   A
MACROS    Z        DECMIC Z                                              MACROS   175   A
MACROS             VFD    12/T."Z"                                       MACROS   176   A
MACROS    Z        SET    Z+1                                            MACROS   177   A
MACROS   *                         TERMINATE LAST WORD OF RELOC BYTES    MACROS   178   A
MACROS             IFNE   *P,60,1                                        MACROS   179   A
MACROS             VFD    *P/0                                           MACROS   180   A
MACROS             USE    INDEXT                                         MACROS   181   A
MACROS             VFD    6/,18/N.PR,18/.LEN,18/.FWA-1                   MACROS   182   A
MACROS             USE    SKELS                                          MACROS   183   A
MACROS    .FWA     SET    0                                              MACROS   184   A
MACROS             ENDM                                                  MACROS   185   A
MACROS                                                                   MACROS   186   A
MACROS    MAX.LM   SET    0          MAX *SMACRO* LENGTH                 MACROS   187   A
MACROS    FRB?     SPACE  2,14                                           MACROS   188   A
MACROS   *         FRB - FORM A RELOC BYTE ( INTERNAL MACRO )            MACROS   189   A
MACROS   *                                                               MACROS   190   A
MACROS   *         FRB?   TYP,BOTTOM-BIT                                 MACROS   191   A
MACROS                                                                   MACROS   192   A
MACROS    FRB?     MACRO  T,BOT                                          MACROS   193   A
MACROS             IFC    EQ,/T/L/,2                                     MACROS   194   A
MACROS    .LB      SET    .LB+1                                          MACROS   195   A
MACROS             SKIP   6                                              MACROS   196   A
MACROS    N.PR     SET    N.PR+1                                         MACROS   197   A
MACROS    Z        DECMIC N.PR                                           MACROS   198   A
MACROS             NOREF  PR."Z"                                         MACROS   199   A
MACROS             ERRPL  *-L.PR-64                                      MACROS   200   A
MACROS    PR."Z"   SET    ;BS6+*-L.PR                                    MACROS   201   A
MACROS    L.PR     SET    *                                              MACROS   202   A
MACROS             ENDM                                                  MACROS   203   A
MACROS                                                                   MACROS   204   A
MACROS             NOREF  .LB                                            MACROS   205   A
MACROS    BTW      SPACE  3,14                                           MACROS   206   A
MACROS   **        BTW - BEGIN TEXT WORD DEFINITION                      MACROS   207   A
MACROS   *         BTW BEGINS THE DEFINITION OF A WORD OF DATA WHOSE FIELDS MAY  MACROS   208   A
MACROS   *         BE FILLED IN WITH VALUES TAKEN FROM THE MACRO PARAMS OR SET   MACROS   209   A
MACROS   *         DEFINED VALUES OR STRING VALUES.                      MACROS   210   A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MACROS

```
MACROS   *         ARI, ACI, MIC, ASV, SET. IF.XX, ELSE. AND ENDIF. MAY APPEAR          MACROS     211    A
MACROS   *         INSIDE A BTW/ETW PAIR.  A BTW FORCES UPPER.                          MACROS     212    A
MACROS   *                                                                              MACROS     213    A
MACROS   *         BTW    ADDR                                                          MACROS     214    A
MACROS   *         ADDR = ADDRESS OF TEMPLATE WORD, MAY NOT BE A LITERAL.               MACROS     215    A
MACROS   *                IF ADDR IS ABSENT, THEN A ZERO TEMPLATE WORD IS ASSUMED       MACROS     216    A
MACROS                                                                                  MACROS     217    A
MACROS    BTW      MACRO  ADDR                                                          MACROS     218    A
MACROS             LOCAL  ETWA                                                          MACROS     219    A
MACROS    BTW      MICRO  1,,/ETWA/                                                     MACROS     220    A
MACROS             ERRNZ  .VFD        BTW/ETW NESTING ERROR                             MACROS     221    A
MACROS             IFC    NE,$ADDR$$,2                                                  MACROS     222    A
MACROS             VFD    12/1023+OC.BTW,18/ETWA,12/,18/ADDR                            MACROS     223    A
MACROS             SKIP   1                                                             MACROS     224    A
MACROS             VFD    12/1023+OC.BTW,18/ETWA,12/,18/PBUF                            MACROS     225    A
MACROS    .VFD     SET    *-.FWA                                                        MACROS     226    A
MACROS             ENDM                                                                 MACROS     227    A
MACROS                                                                                  MACROS     228    A
MACROS   **        ETW - END TEXT WORD DEFINITION                                       MACROS     229    A
MACROS   *                                                                              MACROS     230    A
MACROS   *         ETW                                                                  MACROS     231    A
MACROS                                                                                  MACROS     232    A
MACROS    ETW      MACRO                                                                MACROS     233    A
MACROS             ERRZR  .VFD        BTW/ETW NESTING ERROR                             MACROS     234    A
MACROS             ERRPL  .IFB-.VFD   BTW/ETW MUST BE INSIDE A IF/ELSE/ENDIF            MACROS     235    A
MACROS    "BTW"    EQU    *-1-.FWA    BTW/ETW NESTING ERROR                             MACROS     236    A
MACROS    BTW      MICRO  1,,                                                           MACROS     237    A
MACROS    .VFD     SET    0                                                             MACROS     238    A
MACROS             ENDM                                                                 MACROS     239    A
MACROS    ARI      SPACE  2,14                                                          MACROS     240    A
MACROS   **        ARI - ADD RELOCATION INFO TO TEXT FIELD                              MACROS     241    A
MACROS   *         ACI - ADD CONSTANT VALUE TO TEXT FIELD.                              MACROS     242    A
MACROS   *                                                                              MACROS     243    A
MACROS   *         ARI    TOP-BIT,BIT-LEN,SYMORD                                        MACROS     244    A
MACROS   *  OR     ARI    RELOC,SYMORD/PARAM   RELOC = U, M OR L FOR UPPER,             MACROS     245    A
MACROS   *         MIDDLE OR LOWER RELOCATION.                                          MACROS     246    A
MACROS   *         THE CORRESPONDING *COMPASS* STATEMENTS WOULD BE -                    MACROS     247    A
MACROS   *         POS    TOP-BIT                                                       MACROS     248    A
MACROS   *         VFD    BIT-LEN/PARAM-VALUE  IS ADDED TO THE CURRENT BTW WORD         MACROS     249    A
MACROS                                                                                  MACROS     250    A
MACROS    RL.U     EQU    30          BASE BIT OF RELOCATABLE FIELD                     MACROS     251    A
MACROS    RL.M     EQU    15                                                            MACROS     252    A
MACROS    RL.L     EQU    0                                                             MACROS     253    A
MACROS                                                                                  MACROS     254    A
MACROS             ECHO   ,OCV=(ARI,ACI,AAC)                                            MACROS     255    A
MACROS    OCV      MACRO  TB,LEN,ORD                                                    MACROS     256    A
MACROS             ERRZR  .VFD        OCV MUST BE INSIDE A BTW/ETW PAIR                 MACROS     257    A
MACROS    Z        IFC    EQ,/ORD//                                                     MACROS     258    A
MACROS    .L       SET    18                                                            MACROS     259    A
MACROS    .F       SET    RL.TB+1S10                                                    MACROS     260    A
MACROS    .S       MICRO  1,,/LEN/                                                      MACROS     261    A
MACROS    Z        ELSE                                                                 MACROS     262    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MACROS

```
MACROS    .L       SET    LEN                                                    MACROS     263    A
MACROS    .F       SET    TB+1-.L+1S10                                           MACROS     264    A
MACROS    .S       MICRO  1,,/ORD/                                               MACROS     265    A
MACROS    Z        ENDIF                                                         MACROS     266    A
MACROS             VFD    12/1023+OC.OCV,18/59-.L,12/.F,18/".S"                  MACROS     267    A
MACROS             ENDM                                                          MACROS     268    A
MACROS             ENDD                                                          MACROS     269    A
MACROS                                                                           MACROS     270    A
MACROS             NOREF  .F,.L                                                  MACROS     271    A
MACROS    AAE      SPACE  3,14                                                   MACROS     272    A
MACROS   **        AAE - ADD ADDRESS EXPRESSION TO A TEXT FIELD                  MACROS     273    A
MACROS   *                                                                       MACROS     274    A
MACROS   *         AAE    TOP-BIT,BIT-LEN,SYMORD,CA                              MACROS     275    A
MACROS   *         AAE    RELOC,SYMORD,CA                                        MACROS     276    A
MACROS   *                                                                       MACROS     277    A
MACROS   *         IF SYMORD = CON. , THEN THE VALUE OF THE CA WILL BE CHANGED   MACROS     278    A
MACROS   *         TO THE VALUE OF CUT(CA) AND * IH+CA* WILL BE ADDED TO THE     MACROS     279    A
MACROS   *         FIELD.                                                        MACROS     280    A
MACROS                                                                           MACROS     281    A
MACROS    AAE      MACRO  TB,LEN,ORD,CA                                          MACROS     282    A
MACROS             IFC    EQ,/CA//,3                                             MACROS     283    A
MACROS             ARI    TB,LEN                                                 MACROS     284    A
MACROS             AAC    TB,ORD                                                 MACROS     285    A
MACROS             ELSE   2                                                      MACROS     286    A
MACROS             ARI    TB,LEN,ORD                                             MACROS     287    A
MACROS             AAC    TB,LEN,CA                                              MACROS     288    A
MACROS             ENDM                                                          MACROS     289    A
MACROS    MIC      SPACE  2,14                                                   MACROS     290    A
MACROS   **        MIC - DEFINE CHARACTER STRING VALUE OF SYMTAB ORD             MACROS     291    A
MACROS   *                                                                       MACROS     292    A
MACROS   * MI      MICRO  FIRST-CHAR,,$BCD(SYMORD)$         N-CHAR = 0           MACROS     293    A
MACROS   * MI      MICRO  FIRST-CHAR,N-CHAR,$BCD(SYMORD)      $   N-CHAR " 0     MACROS     294    A
MACROS   *         N-CHAR = NUMBER OF CHARACTERS, IF = 0 THEN TAKE CHARS UNTIL   MACROS     295    A
MACROS   *                SEPERATOR CHAR                                         MACROS     296    A
MACROS   *         MIC IS EQUIVALENT TO THE STATEMENT -                          MACROS     297    A
MACROS   * MI      MICRO  FIRST-CHAR,N-CHAR,$BCD(SYMORD)$                        MACROS     298    A
MACROS   *         WHERE $ IS THE SEP-CHAR.                                      MACROS     299    A
MACROS                                                                           MACROS     300    A
MACROS                                                                           MACROS     301    A
MACROS             MACRO  MIC,M,FC,NC,SC,SYMORD                                  MACROS     302    A
MACROS             IF     -DEF,/SYM/SYMORD,1                                     MACROS     303    A
MACROS             ERR    SYMORD MUST BE A SYMTAB ORDINAL                        MACROS     304    A
MACROS             VFD   12/1023+OC.MIC,12/M-M1,6/6*NC,6/6*FC-6,6/1R_SC,18/SYMORD MACROS     305    A
MACROS             ENDM                                                          MACROS     306    A
MACROS    ASV      SPACE  2,14                                                   MACROS     307    A
MACROS   **        ASV - ADD STRING VALUE TO FIELD                               MACROS     308    A
MACROS   *                                                                       MACROS     309    A
MACROS   *         ASV    TOP-BIT,NC,MIC                                         MACROS     310    A
MACROS   *         TOP-BIT = LEFTMOST BIT OF WORD                                MACROS     311    A
MACROS   *         THE STRING NC_L"MIC" IS ADDED TO THE FIELD STARTING AT TOP-BIT MACROS     312    A
MACROS                                                                           MACROS     313    A
MACROS    ASV      MACRO  TB,NC,MIC                                              MACROS     314    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MACROS

```
MACROS            VFD    12/1023+OC.ASV,6/TB,18/6*NC-1,6/,18/MIC                MACROS    315    A
MACROS            ENDM                                                          MACROS    316    A
MACROS    IF.XX   SPACE  3,16                                                   MACROS    317    A
MACROS   **       IF.XX CONDITIONALLY ASSEMBLE MACRO CODE                       MACROS    318    A
MACROS   *        CODE IS SKIPPED UNTIL ENDIF OR ELSE IF P1 XX P2 IS FALSE.     MACROS    319    A
MACROS   *                                                                      MACROS    320    A
MACROS   * LAB    IF.XX  P1,P2    ( XX = LT,LE,GT,GE,EQ, OR NE )                MACROS    321    A
MACROS   *        PI - IF TEST PARAMETERS, ARE INTERPRETED AS CONSTANTS.        MACROS    322    A
MACROS                                                                          MACROS    323    A
MACROS            NOREF  .O1,.O2,.IFB,.VFD                                      MACROS    324    A
MACROS    IFSTK   MICRO  1,,                                                    MACROS    325    A
MACROS                                                                          MACROS    326    A
MACROS            ECHO   ,XX=(EQ,NE,GE,LE,LT,GT)                                MACROS    327    A
MACROS   *                                                                      MACROS    328    A
MACROS            MACRO  IF.XX,LOC,P1,P2                                        MACROS    329    A
MACROS            LOCAL  LABL                                                   MACROS    330    A
MACROS            IFC    EQ,/LOC//,1                                            MACROS    331    A
MACROS            ERR    IF.XX MUST BE LABELED                                  MACROS    332    A
MACROS    LOC.    MICRO  1,,/LABL/                                              MACROS    333    A
MACROS    I       ECHO   ,P=(P1,P2),V=(.O1,.O2)                                 MACROS    334    A
MACROS            IF     -ABS,P,2                                               MACROS    335    A
MACROS    V       SET    P-PBUF                                                 MACROS    336    A
MACROS            SKIP   1                                                      MACROS    337    A
MACROS    V       LITORD? P                                                     MACROS    338    A
MACROS    I       ENDD                                                          MACROS    339    A
MACROS            VFD    12/1023+OC.IFT,3/IF.XX,15/LABL-*,15/.O1,15/.O2         MACROS    340    A
MACROS    .Z      DECMIC .IFB,5                                                 MACROS    341    A
MACROS    IFSTK   MICRO  1,,/".Z""IFSTK"/                                       MACROS    342    A
MACROS    .IFB    SET    *-.FWA                                                 MACROS    343    A
MACROS            ENDM                                                          MACROS    344    A
MACROS            ENDD                                                          MACROS    345    A
MACROS    ELSE    SPACE  2,10                                                   MACROS    346    A
MACROS   **       ELSE. - REVERSE EFFECTS OF *IF.XX*                            MACROS    347    A
MACROS   *                                                                      MACROS    348    A
MACROS   * LAB    ELSE.                                                         MACROS    349    A
MACROS                                                                          MACROS    350    A
MACROS            MACRO  ELSE.,LOC                                              MACROS    351    A
MACROS            LOCAL  LABL                                                   MACROS    352    A
MACROS            ERRMI  .IFB-.VFD   BTW/ETW MUST BE CONTAINED WITH A IF/ELSE   MACROS    353    A
MACROS   *                           DEFINE LOC FOR IF SKIP COUNT               MACROS    354    A
MACROS    "LOC."  BSS    0                                                      MACROS    355    A
MACROS    LOC.    MICRO  1,,/LABL/                                              MACROS    356    A
MACROS            VFD    12/1023+OC.ELSE,30/,18/LABL-*                          MACROS    357    A
MACROS            ENDM                                                          MACROS    358    A
MACROS    ENDIF.  SPACE  2,10                                                   MACROS    359    A
MACROS   **       ENDIF. - TERMINATE IF RANGE                                   MACROS    360    A
MACROS   *                                                                      MACROS    361    A
MACROS   * LAB    ENDIF.                                                        MACROS    362    A
MACROS                                                                          MACROS    363    A
MACROS            MACRO  ENDIF.,LOC                                             MACROS    364    A
MACROS            ERRMI  .IFB-.VFD   BTW/ETW MUST BE CONTAINED WITH A IF/ELSE   MACROS    365    A
MACROS    "LOC."  EQU    *-1                                                    MACROS    366    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MACROS

```
MACROS    IFSTK    MICRO  6,,/"IFSTK"/                                    MACROS    367    A
MACROS    .Z       MICRO  1,5,/"IFSTK"/                                   MACROS    368    A
MACROS    .IFB     SET    ".Z"                                            MACROS    369    A
MACROS             ENDM                                                   MACROS    370    A
MACROS    DCSS     SPACE  2,12                                            MACROS    371    A
MACROS   **        DCSS - DEFINE COMPILER SPECIAL SYMBOL THAT IS GLOBAL   MACROS    372    A
MACROS   *         ( REFERENCED IN DIFFERENT MACROS ).                    MACROS    373    A
MACROS   *         A *DCSS* DEFINED SYMBOL MAY BE DEFINED BY A SET. STATEMENT  MACROS  374  A
MACROS   *         AND REFERENCED IN OTHER SET. OR IF.XX STATEMENTS.      MACROS    375    A
MACROS   *                                                                MACROS    376    A
MACROS   * NAME    DCSS   INITIAL-VALUE,TYPE                              MACROS    377    A
MACROS   *         INITIAL-VALUE IS ANY ADDRESS EXPRESSION AND IS OPTIONAL.  MACROS  378  A
MACROS   *         TYPE = *SYM* OR *CON* WHICH INDICATES WETHER THE NAME IS TO  MACROS  379  A
MACROS   *         BE INTERPRETED AS A SYMBOL TABLE ORDINAL OR A CONSTANT.  MACROS  380  A
MACROS   *         DEFAULT IS *CON*.                                      MACROS    381    A
MACROS                                                                    MACROS    382    A
MACROS             MACRO  DCSS,NAME,VAL,TYP                               MACROS    383    A
MACROS    NAME     CON    VAL 0                                           MACROS    384    A
MACROS             QUAL   TYP CON                                         MACROS    385    A
MACROS    NAME     EQU    0                                               MACROS    386    A
MACROS             QUAL                                                   MACROS    387    A
MACROS             ENDM                                                   MACROS    388    A
MACROS   SET.      SPACE  2,14                                            MACROS    389    A
MACROS   **        SET. - SET SYMBOL VALUE TO VALUE OF ADDRESS EXPRESSION.  MACROS  390  A
MACROS   *                                                                MACROS    391    A
MACROS   * LN      SET.   A-E                                             MACROS    392    A
MACROS   *         LN - A LOCAL SYMBOL ( L1,L2,... )                      MACROS    393    A
MACROS   *         A-E - AN ADDRESS EXPRESSION WHICH MAY CONTAIN CONSTANTS,  MACROS  394  A
MACROS   *         PARAMETERS, SYMBOL ORDINALS OR * ( THE ORG COUNTER ) AS TERMS,  MACROS  395  A
MACROS   *         AND + - * AND / AS OPERATORS. EVALUATION IS FROM LEFT TO RIGHT  MACROS  396  A
MACROS   *         WARNING - THERE IS VERY LITTLE ERROR CHECKING, SO BE CAREFUL.  MACROS  397  A
MACROS   *         LET *ANY* = ABS OR REL, THEN ABS-REL, ANY*REL, ANY/REL AND  MACROS  398  A
MACROS   *         REL/ANY ARE ILLEGAL OPERATOR OPERAND COMBINATIONS.     MACROS    399    A
MACROS                                                                    MACROS    400    A
MACROS             MACRO  SET.,LOC,AE                                     MACROS    401    A
MACROS   *                        INITIALIZE                              MACROS    402    A
MACROS    LV       MICRO  1,,/LOC/                                        MACROS    403    A
MACROS    AES      MICRO  1,,$AE $                                        MACROS    404    A
MACROS    Z        SET    1          STRING POINTER                       MACROS    405    A
MACROS             GETTKN OP1                                             MACROS    406    A
MACROS             IFC    EQ,$"OP1"$*$,1                                  MACROS    407    A
MACROS    OP1      MICRO  1,,/ORG#CTR/                                    MACROS    408    A
MACROS   *                        CHECK FOR LEADING UNARY MINUS           MACROS    409    A
MACROS             IFC    EQ,$"OP1"$-$,4                                  MACROS    410    A
MACROS             GETOPND OP2                                            MACROS    411    A
MACROS    OP1      MICRO  1,,/PBUF/                                       MACROS    412    A
MACROS             OUTSET -                                               MACROS    413    A
MACROS    SET      IFGE   Z,1                                             MACROS    414    A
MACROS   *                        LOOP TO EVAL REST OF EXPRESSION         MACROS    415    A
MACROS             DUP    50                                              MACROS    416    A
MACROS             GETOPER                                                MACROS    417    A
MACROS             GETOPND OP2                                            MACROS    418    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MACROS

```
MACROS              OUTSET "OP"                                          MACROS     419    A
MACROS    OP1       MICRO  1,,/L0/                                       MACROS     420    A
MACROS              IFEQ   Z,0,1                                         MACROS     421    A
MACROS              STOPDUP                                              MACROS     422    A
MACROS              ENDD                                                 MACROS     423    A
MACROS    SET       ENDIF                                                MACROS     424    A
MACROS              ENDM                                                 MACROS     425    A
MACROS                                                                   MACROS     426    A
MACROS    GETOPND   MACRO  OP                                            MACROS     427    A
MACROS    OP        MICRO  1,,/PBUF/                                     MACROS     428    A
MACROS    W         IFNE   Z,0                                           MACROS     429    A
MACROS              GETTKN OP                                            MACROS     430    A
MACROS              IFC    EQ,$"OP"$*$,1                                 MACROS     431    A
MACROS    OP        MICRO  1,,/ORG#CTR/                                  MACROS     432    A
MACROS    Z         MICRO  1,1,$"OP"$                                    MACROS     433    A
MACROS              IFC    GT,$"Z"$9$,1                                  MACROS     434    A
MACROS              ERR    SET/EQU EXPECTED NAME, GOT - $"OP"$           MACROS     435    A
MACROS              IF     ABS,"OP",2                                    MACROS     436    A
MACROS    ."OP"     LITORD? "OP"                                         MACROS     437    A
MACROS    OP        MICRO  1,,$PBUF+."OP"$                               MACROS     438    A
MACROS    W         ENDIF                                                MACROS     439    A
MACROS              ENDM                                                 MACROS     440    A
MACROS                                                                   MACROS     441    A
MACROS    GETOPER   MACRO                                                MACROS     442    A
MACROS    OP        MICRO  1,,/+/                                        MACROS     443    A
MACROS              IFNE   Z,0,3                                         MACROS     444    A
MACROS              GETTKN OP                                            MACROS     445    A
MACROS              IFC LT,$"OP"$+$,1                                    MACROS     446    A
MACROS              ERR    SET/EQU EXPECTED AN OPERATOR, GOT $"OP"$      MACROS     447    A
MACROS              ENDM                                                 MACROS     448    A
MACROS                                                                   MACROS     449    A
MACROS     OUTSET   MACRO  OP                                            MACROS     450    A
MACROS    *         ASSUMES OP1,OP2 SETUP, Z=0 IF NOMORE                 MACROS     451    A
MACROS     RES      MICRO  1,,/L0/                                       MACROS     452    A
MACROS              IFEQ   Z,0,1                                         MACROS     453    A
MACROS     RES      MICRO  1,,/"LV"/                                     MACROS     454    A
MACROS              ECHO   3,AR=("RES","OP2","OP1"),RV=(.R1,.O2,.O1)     MACROS     455    A
MACROS     RV       SET    AR-PBUF                                       MACROS     456    A
MACROS              IF     DEF,/SYM/AR,1                                 MACROS     457    A
MACROS     RV       SET    RV+1S14                                       MACROS     458    A
MACROS              VFD    12/1023+OC.SET,15/.R1,3/"OP",15/.O2,15/.O1    MACROS     459    A
MACROS              ENDM                                                 MACROS     460    A
MACROS                                                                   MACROS     461    A
MACROS     GETTKN   MACRO  OP          RETURN NEXT TOKEN IN MICRO $OP$  MACROS     462    A
MACROS     ZB       SET    Z                                             MACROS     463    A
MACROS    *                            CHECK FIRST CHARACTER FOR OP OR NAME MACROS 464    A
MACROS     OP       MICRO  Z,1,$"AES"$                                   MACROS     465    A
MACROS     X        IFC    GT,$"OP"$9$                                   MACROS     466    A
MACROS     Z        SET    Z+1                                           MACROS     467    A
MACROS              IFC    EQ,$"OP"$ $,1                                 MACROS     468    A
MACROS     Z        SET    0                                             MACROS     469    A
MACROS     X        ELSE                                                 MACROS     470    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MACROS

```
MACROS   *                          ASSEMBLE A NAME                          MACROS      471    A
MACROS            DUP    8                                                    MACROS      472    A
MACROS    Z       SET    Z+1                                                  MACROS      473    A
MACROS    Z       MICRO  Z,1,$"AES"$                                          MACROS      474    A
MACROS            IFC    GT,$"Z"$9$,5                                         MACROS      475    A
MACROS            IFC    NE,$"Z"$.$,4                                         MACROS      476    A
MACROS    OP      MICRO  ZB,Z-ZB,$"AES"$                                      MACROS      477    A
MACROS            IFC    EQ,$"Z"$ $,1                                         MACROS      478    A
MACROS    Z       SET    0                                                    MACROS      479    A
MACROS            STOPDUP                                                     MACROS      480    A
MACROS            ENDD                                                        MACROS      481    A
MACROS    X       ENDIF                                                       MACROS      482    A
MACROS            ENDM                                                        MACROS      483    A
MACROS                                                                        MACROS      484    A
MACROS    PSEUDOS SPACE  3,14                                                 MACROS      485    A
MACROS   **       THE PSEUDOS DEFINED BELOW MAY APPEAR IN *SMACROS* WITH THE  MACROS      486    A
MACROS   *        RESTRICTION THAT THEY NOT APPEAR IN BETWEEN A BTD/ETD OR    MACROS      487    A
MACROS   *        BTW/ETW PAIR.                                               MACROS      488    A
MACROS    USE     SPACE  2,10                                                 MACROS      489    A
MACROS   **       USE. - CHANGE USE BLOCKS                                    MACROS      490    A
MACROS   *                                                                    MACROS      491    A
MACROS   *        USE.   BN   , BN = FIXED LOCAL BLOCK NAME OR B-PARAM        MACROS      492    A
MACROS   *                          CLF = CONDITIONAL LOAD FLAG ( BLANK OR 1 ) CCGA005       45    A
MACROS                                                                        MACROS      493    A
MACROS    USE.    MACRO  BN                                                   MACROS      494           I
         -CCGA005
MACROS    USE.    MACRO  BN,CLF                                               CCGA005      46    A
MACROS            ERRNZ  .VFD        USE MAY NOT APPEAR INSIDE A BTW/ETW PAIR MACROS      495    A
MACROS            IF     -DEF,BN$,2                                           MACROS      496    A
MACROS            ERR    ILLEGAL PARAMETER TO USE. MACRO - BN                 MACROS      497    A
MACROS            SKIP   1                                                    MACROS      498    A
MACROS    .Z      SET    BN$                                                  MACROS      499    A
MACROS            IF     REG,BN,1                                             MACROS      500    A
MACROS            FRB?   B,0                                                  MACROS      501    A
MACROS            VFD    12/1023+OC.USE,48/.Z                                 MACROS      502           I
         -CCGA005
MACROS            VFD    12/1023+OC.USE,23/,1/CLF,24/.Z                       CCGA005      47    A
MACROS            ENDM                                                        MACROS      503    A
MACROS    QUAL    SPACE  2,14                                                 MACROS      504    A
MACROS   **       QUAL.  CHANGE QUAL BLOCK NAME                               MACROS      505    A
MACROS   *                                                                    MACROS      506    A
MACROS   *        QUAL   QN , QN = BLANK , * OR A Q-PARAM                     MACROS      507    A
MACROS                                                                        MACROS      508    A
MACROS    QUAL.   MACRO  QN                                                   MACROS      509    A
MACROS    .Z      SET    0                                                    MACROS      510    A
MACROS            IFC    NE,/QN//                                             MACROS      511    A
MACROS    .Z      SET    1                                                    MACROS      512    A
MACROS            IFC    NE,/QN/*/                                            MACROS      513    A
MACROS    Z       MICRO  1,1,/QN/                                             MACROS      514    A
MACROS            IFC    NE,/"Z"/Q/,1                                         MACROS      515    A
MACROS            ERR    ILLEGAL ARG TO QUAL. - QN                            MACROS      516    A
MACROS    .Z      SET    QN                                                   MACROS      517    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MACROS

```
MACROS            ENDIF                                                  MACROS      518    A
MACROS            IF     -ABS,.Z,1                                       MACROS      519    A
MACROS            FRB?   Q,0                                             MACROS      520    A
MACROS            VFD    12/1023+OC.QUAL,48/.Z                           MACROS      521    A
MACROS            ENDM                                                   MACROS      522    A
MACROS    CON     SPACE  2,10                                            MACROS      523    A
MACROS   **       CON. - DEFINE A RELOCATABLE WORD OF DATA               MACROS      524    A
MACROS   *                                                               MACROS      525    A
MACROS   * LAB    CON.   SY,CA                                           MACROS      526    A
MACROS                                                                   MACROS      527    A
MACROS            MACRO  CON.,LAB,SY,CA                                  MACROS      528    A
MACROS            ERRNZ  .VFD        CON MAY NOT APPEAR INSIDE A BTW/ETW PAIR  MACROS  529    A
MACROS            IFC    NE,/LAB//,1                                     MACROS      530    A
MACROS    LAB     BSS.   0                                               MACROS      531    A
MACROS    .LB     SET    0                                               MACROS      532    A
MACROS            ECHO   4,T=(SY,CA),P=(30,0)                            MACROS      533    A
MACROS    .LB     SET    2*.LB                                           MACROS      534    A
MACROS            IF     -ABS,T,2                                        MACROS      535    A
MACROS    Z       MICRO  1,1,/T/                                         MACROS      536    A
MACROS            FRB?   "Z",P                                           MACROS      537    A
MACROS            VFD    12/1023+OC.CON,18/SY,6/.LB,24/CA                MACROS      538    A
MACROS            ENDM                                                   MACROS      539    A
MACROS    BSS     SPACE  2,14                                            MACROS      540    A
MACROS   **       BSS./BSSZ. - RESERVE STORAGE SPACE.                    MACROS      541    A
MACROS   *                                                               MACROS      542    A
MACROS   * LAB    BSS.   WC                                              MACROS      543    A
MACROS                                                                   MACROS      544    A
MACROS            ECHO   ,OCV=(BSS,BSSZ)                                 MACROS      545    A
MACROS            MACRO  OCV.,LOC,WC                                     MACROS      546    A
MACROS            ERRNZ  .VFD        OCV MAY NOT APPEAR INSIDE A BTW/ETW PAIR  MACROS  547    A
MACROS    .LB     SET    0                                               MACROS      548    A
MACROS            IF     -ABS,WC,2                                       MACROS      549    A
MACROS    Z       MICRO  1,1,/WC/                                        MACROS      550    A
MACROS            FRB?   "Z",0                                           MACROS      551    A
MACROS            VFD    12/1023+OC.OCV,24/.LB,24/WC                     MACROS      552    A
MACROS            IFC    NE,/LOC//,1                                     MACROS      553    A
MACROS    LOC     SET.   *-WC                                            MACROS      554    A
MACROS            ENDM                                                   MACROS      555    A
MACROS            ENDD                                                   MACROS      556    A
MACROS    SMACROS TITLE  HOST COMPILER DEFINED STORAGE MACROS            MACROS      557    A
MACROS   *CALL    SMACROS                                                MACROS      558    A
MACROS                                                                   MACROS      559    A
MACROS   *        MACRO EXPANSION BUFFER                                 MACROS      560    A
MACROS                                                                   MACROS      561    A
MACROS            USE    /CCG.SCR/                                       MACROS      562         I
         -CCG18
MACROS            USE    /CCGSCR/                                        CCG18        10    A
MACROS    F$MXB   ENTRY. 0                                               MACROS      563    A
MACROS            BSS    MAX.LM                                          MACROS      564    A
MACROS    .US     SET    F$MXB+260-*       UNUSED SPACE                  MACROS      565    A
MACROS            IFGT   .US,0,1                                         MACROS      566    A
MACROS            BSS    .US                                             MACROS      567    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MACROS

MACROS              USE    0                                                    MACROS        568     A
MACROS                                                                          MACROS        569     A
MACROS              END                                                          MACROS        570     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - MACROS

          IDENTIFIER                TOTAL    ACTIVE

          MACROS                      570       567
          CCG18                         1         1
          CCGA005                       3         3

          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

CGTM      *DECK     CGTM                                                         CGTM            1     A
CGTM                IDENT  CGTM                                                  CGTM            2     A
CGTM       CGTM     TITLE  CGTM - CODE GENERATOR TABLE MANAGER                   CGTM            3     A
CGTM       B=CGTM   RPVDEF                                                       CGTM            4     A
CGTM      *CALL     SSTCALL                                                      CGTM            5     A
CGTM                SPACE  2                                                     CCG*78          4     A
CGTM      **        CONTROL DATA PROPRIETARY PRODUCT                             CCG*78          5           I
          -CCG1*92
CGTM      *         COPYRIGHT CONTROL DATA CORPORATION 1978.                     CCG*78          6           I
          -CCG*80
CGTM      *     COPYRIGHT CONTROL DATA CORP. 1978, 1979, 1980                    CCG*80          4           I
          -CCG*81
CGTM      *         COPYRIGHT CONTROL DATA CORP. 1978, 1979, 1980, 1981.
>>>>      CCG*81  10                                                             CCG*81          4           I
          -CCG*82
CGTM      *         COPYRIGHT CONTROL DATA CORP. 1978, 1979, 1980, 1981, 1982.   CCG*82          4           I
          -CCG1*92
CGTM      *         COPYRIGHT CONTROL DATA SYSTEMS INC. 1992                     CCG1*92        10           I
          -CCG*94
CGTM      *         COMMENT COPYRIGHT CONTROL DATA SYSTEMS, INC. 1994.           CCG*94         12     A
CGTM                SPACE  2                                                     CCG*78          7     A
CGTM                                                                             CGTM            6     A
CGTM      **        CGTM - CODE GENERATOR TABLE MANAGER AND UTILITY ROUTINES     CGTM            7     A
CGTM       OPT      SPACE  3                                                     CGTM            8     A
CGTM      **        F.OPT - OPT=2 RANDOM FILE FET AND FIT                        CGTM            9     A
CGTM                                                                             CGTM           10     A
CGTM                ENTRY  F.OPT                                                 CGTM           11     A
CGTM       F.OPT    BSS    0                                                     CGTM           12     A
CGTM                                                                             CGTM           13     A
CGTM       #RM      IFEQ   CP#RM,0                                               CGTM           14     A
CGTM                                                                             CGTM           15     A
CGTM                VFD    42/7LZZZZZOP,18/3                                     CGTM           16     A
CGTM                VFD    42/3,18/0                                             CGTM           17     A
CGTM                BSSZ   2                                                     CGTM           18     A
CGTM                CON    200B                                                  CGTM           19     A
CGTM                BSSZ   3                                                     CGTM           20     A
CGTM       #RM      ELSE                                                         CGTM           21     A
CGTM                                                                             CGTM           22     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM      FF.OPT   FILE   LFN=ZZZZZOP,FO=WA,OF=R,CF=U,PD=IO,ET=T,RT=W,MRL=377777B        CGTM          23          I
         -CCGA018
CGTM               ENTRY  FI.OPT                                                        CCGA018      116     A
CGTM      FI.OPT   BSS    0                                                              CCGA018      117     A
CGTM      FF.OPT   FILE   LFN=ZZZZZOP,FO=WA,OF=R,CF=U,PD=IO,EO=T,RT=W,MRL=377777B        CCGA018      118     A
CGTM               BSSZ   FF.OPT+20D-*                                                   CGTM          24     A
CGTM               ENTRY  FF.OPT                                                         CGTM          25     A
CGTM                                                                                     CGTM          26     A
CGTM      #RM      ENDIF                                                                 CGTM          27     A
CGTM      CONTV    SPACE  3,14                                                           CCGA072       27     A
CGTM      .T       IFNE   TEST,0                                                         CCGA072       28     A
CGTM     **        CONTV - RETURN I TH VALUE FROM THE CON TABLE                          CCGA072       29     A
CGTM                                                                                     CCGA072       30     A
CGTM      CONTV    ENTRY.                                                                CCGA072       31     A
CGTM               SA1    X1                                                             CCGA072       32     A
CGTM               SA2    =XO$CVT                                                        CCGA072       33     A
CGTM               IX3    X1+X2                                                          CCGA072       34     A
CGTM               SA4    X3          CVT(ARG)                                           CCGA072       35     A
CGTM               BX6    X4                                                             CCGA072       36     A
CGTM               EQ     CONTV                                                          CCGA072       37     A
CGTM      .T       ENDIF                                                                 CCGA072       38     A
CGTM      CG$IEP   TITLE  UTILITY ROUTINES                                               CGTM          28     A
CGTM     **        CG$IEP - INITIALIZE END PROCESSING                                    CGTM          29     A
CGTM     *         CLOSE -OPT- FILE, ADJUST WORKING STORAGE LIMITS, ADJUST FL.           CGTM          30     A
CGTM                                                                                     CGTM          31     A
CGTM      CG$IEP   ENTRY.                                                                CGTM          32     A
CGTM               SETB1                                                                 CGTM          33     A
CGTM               SX6    TOV#                                                           CGTM          34     A
CGTM               SA6    TO          TO = TOV#     */ RESTORE TBL MGR EXIT              CGTM          35     A
CGTM               SX6    =XHE$EPX                                                       CGTM          36     A
CGTM               SA6    CG.MOX      SET END PROCESSING OVERFLOW EXIT                   CGTM          37     A
CGTM               SA1    =XHO$OPT                                                       CGTM          38     A
CGTM               PL     X1,IEP1     IF CO.OPT " 2                                      CGTM          39     A
CGTM                                                                                     CGTM          40     A
CGTM     *         CLOSE THE OPT FILE                                                    CGTM          41     A
CGTM                                                                                     CGTM          42     A
CGTM      #RM      IFEQ   CP#RM,0                                                        CGTM          43     A
CGTM               SX6    1000B                                                          CGTM          44     A
CGTM               SA6    =XF.OPT+1   CLEAR RANDOM BIT, SET POINTERS TO                  CGTM          45     A
CGTM               SA6    A6+B1       VALUE THAT IS < FL                                 CGTM          46     A
CGTM               SA6    A6+B1                                                          CGTM          47     A
CGTM               LX6    1                                                              CGTM          48     A
CGTM               SA6    A6+B1       LIMIT                                              CGTM          49     A
CGTM               CLOSE  F.OPT,UNLOAD      RETURN THE FILE                               CGTM          50          I
         -CCG0187
CGTM               CLOSE  F.OPT,UNLOAD,R    RETURN THE FILE                               CCG0187        8     A
CGTM      #RM      ELSE                                                                  CGTM          51     A
CGTM               CLOSEM =XFF.OPT,U                                                     CGTM          52     A
CGTM      #RM      ENDIF                                                                 CGTM          53     A
CGTM                                                                                     CGTM          54     A
CGTM      IEP1     SX6    =XB=MIO     SET NEW LOW MEMORY LIMIT                           CGTM          55     A
CGTM               SA6    LM                                                             CGTM          56     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM              SA6    O$BLK                                                  CGTM        57    A
CGTM              SA6    O$TXT                                                  CGTM        58    A
CGTM              EQ     CG$IEP                                                 CGTM        59    A
CGTM      PUNT    SPACE  3,14                                                   CGTM        60    A
CGTM     **       PUNT - TERMINATE CCG OVERLAY PROCESSING WHEN INSUFFICIENT     CGTM        61    A
CGTM     *                MEMORY IS AVAILABLE TO COMPILE.                       CGTM        62    A
CGTM     *                                                                      CGTM        63    A
CGTM     *        ENTRY  (X1) = 7L_CHARS                                        CGTM        64    A
CGTM                                                                            CGTM        65    A
CGTM      PUNTA   DIS    ,$ CCG - INSUFFICIENT MEMORY TO COMPILE - 1234567$     CGTM        66    A
CGTM                                                                            CGTM        67    A
CGTM      PUNT    ENTRY. **,#                                                   CGTM        68    A
CGTM              BX6    X1                                                     CGTM        69    A
CGTM              SA6    PUNTA+4                                                CGTM        70    A
CGTM              IFNE   TEST,0,1                                               CGTM        71    A
CGTM              MESSAGE PUNTA,,RCL * CCG - INSUFFICIENT MEM TO COMPILE *      CGTM        72    A
CGTM              SA2    =XN$FERR                                               CGTM        73    A
CGTM              SX7    X2+B1       N.FERR = N.FERR + 1                        CGTM        74    A
CGTM              SA7    A2                                                     CGTM        75    A
CGTM              SA3    CG.MOX                                                 CGTM        76    A
CGTM              SB2    X3+                                                    CGTM        77    A
CGTM              JP     B2          EXIT TO CONTROL ROUTINE                    CGTM        78    A
CGTM                                                                            CGTM        79    A
CGTM      CG.MOX  ENTRY. =XHE$CTX,#  MEMORY OVERFLOW EXIT                       CGTM        80    A
CGTM      CG$SCT  SPACE  3,14                                                   CGTM        81    A
CGTM     **       CG$SCT - SEARCH CON VALUE TABLE                               CGTM        82    A
CGTM     *                                                                      CGTM        83    A
CGTM     *        ENTRY  (X1) = BINARY CONSTANT TO BE ADDED                     CGTM        84    A
CGTM     *                                                                      CGTM        85    A
CGTM     *        EXIT   (X6) = ORDINAL OF CONSTANT IN THE TABLE                CGTM        86    A
CGTM                                                                            CGTM        87    A
CGTM      SCT0    ADDWRD CUT,B0                                                 CGTM        88    A
CGTM              SX6    X3-1                                                   CGTM        89    A
CGTM                                                                            CGTM        90    A
CGTM      CG$SCT  ENTRY.                                                        CGTM        91    A
CGTM              SETB1                                                         CGTM        92    A
CGTM              ADDWRD CVT,X1      ADDWRD( CVT , CON )                        CGTM        93    A
CGTM              SA4    X2+                                                    CGTM        94    A
CGTM              SB7    X2+1                                                   CGTM        95    A
CGTM      SCT1    BX1    X4-X6                                                  CGTM        96    A
CGTM              SA4    A4+B1                                                  CGTM        97    A
CGTM              NZ     X1,SCT1     IF CON " CVT(I)                            CGTM        98    A
CGTM              MI     X1,SCT1     IF NOT COMPLEMENTS                         CGTM        99    A
CGTM                                                                            CGTM       100    A
CGTM              SB7    A4-B7                                                  CGTM       101    A
CGTM              SX6    B7                                                     CGTM       102    A
CGTM              EQ     B7,B6,SCT0  IF MATCH ON LAST ENTRY                     CGTM       103    A
CGTM              SX7    B6                                                     CGTM       104    A
CGTM              SA6    A3          L.CVT = OLD(L.CVT)                         CGTM       105          I
         -CCG18
CGTM              SA7    A3          L.CVT = OLD(L.CVT)                         CCG18       11    A
CGTM              EQ     CG$SCT                                                 CGTM       106    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM      ENC      SPACE  3,14                                                  CCG18        12     A
CGTM     **        ENC - ENTER N-WORD CONSTANT                                  CCG18        13     A
CGTM     *                                                                      CCG18        14     A
CGTM     *         ENTRY  (X0) = TBL , ADDRESS OF VECTOR OF CONSTANTS           CCG18        15     A
CGTM     *                (X5) = N , NUMBER OF CONSTANTS                        CCG18        16     A
CGTM     *                                                                      CCG18        17     A
CGTM     *         EXIT   (X6) = ORDINAL OF FIRST CONSTANT IN CVT               CCG18        18     A
CGTM                                                                            CCG18        19     A
CGTM      ENC0     SX6    B7          ORD = I                                   CCG18        20     A
CGTM               SX7    B6                                                    CCG18        21     A
CGTM               SA7    L$CVT       L.CVT = OLEN                              CCG18        22     A
CGTM      CG$ENC   ENTRY.                                                       CCG18        23     A
CGTM               SETB1                                                         CCG18        24     A
CGTM               ALLOC  CVT,X5      ALLOC( CVT , N ) , OLEN = OLD(L.CVT)      CCG18        25     A
CGTM               SB5    X5          (B5) = N                                  CCG18        26     A
CGTM               SB4    X2          (B4) = O.CVT                              CCG18        27     A
CGTM               SX3    X2+B6                                                 CCG18        28     A
CGTM               MOVE   X5,X0,X3    MOVE( N , TBL , CVT(OLEN) )               CCG18        29     A
CGTM               SB7    -1          I = -1                                    CCG18        30     A
CGTM                                                                            CCG18        31     A
CGTM      ENC1     SA4    B4+B7                                                 CCG18        32     A
CGTM               SA3    X0                                                    CCG18        33     A
CGTM               SB3    B0          K = 0                                     CCG18        34     A
CGTM               SB7    B7+B1       I = I + 1                                 CCG18        35     A
CGTM                                                                            CCG18        36     A
CGTM      ENC2     BX6    X3-X4                                                 CCG18        37     A
CGTM               SA4    A4+B1                                                 CCG18        38     A
CGTM               SA3    A3+B1                                                 CCG18        39     A
CGTM               SB3    B3+B1       K = K + 1                                 CCG18        40     A
CGTM               NZ     X6,ENC1     IF CVT(I+5) " TBL(K)                      CCG18        41     A
CGTM               MI     X6,ENC1     IF COMPLEMENTS                            CCG18        42     A
CGTM               LT     B3,B5,ENC2  IF K < N                                  CCG18        43     A
CGTM                                                                            CCG18        44     A
CGTM     *         MATCH FOUND, SEE IF CONSTANT IS IN *OLD* TABLE               CCG18        45     A
CGTM                                                                            CCG18        46     A
CGTM               SB3    B7+B5       LWAC = I + N  */ LWA OF NEW CON IN CVT    CCG18        47     A
CGTM               LE     B3,B6,ENC0  IF LWAC @ OLEN                            CCG18        48     A
CGTM                                                                            CCG18        49     A
CGTM     *         EXPAND *CVT*                                                 CCG18        50     A
CGTM                                                                            CCG18        51     A
CGTM               SX7    B3                                                    CCG18        52     A
CGTM               SA7    L$CVT       L.CVT = LWAC                              CCG18        53     A
CGTM               SX5    B7                                                    CCG18        54     A
CGTM               ALLOC  CUT,B3-B6   ALLOC( CUT , LWAC-OLEN )                  CCG18        55     A
CGTM               SETZERO X2+B6,X1                                             CCG18        56     A
CGTM               BX6    X5          ORD = I                                   CCG18        57     A
CGTM               EQ     CG$ENC                                                CCG18        58     A
CGTM      CG$FCU   SPACE  3,14                                                  CGTM        107     A
CGTM     **        CG$FCU - FORCE USE OF K"TH CONSTANT IN *CVT*                 CGTM        108     A
CGTM     *                                                                      CGTM        109     A
CGTM     *         ENTRY  (X1) = ORDINAL OF THE CONSTANT                        CGTM        110     A
CGTM                                                                            CGTM        111     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM      CG$FCU   ENTRY.                                                  CGTM        112     A
CGTM               SA2    O$CUT                                            CGTM        113         I
          -CCG16
CGTM               SA2    =XO$CUT                                          CCG16        64     A
CGTM               SX6    B1                                               CGTM        114     A
CGTM               SB2    X1                                               CGTM        115     A
CGTM               SA6    X2+B2       CUT(K) = 1                           CGTM        116     A
CGTM               EQ     CG$FCU                                           CGTM        117     A
CGTM      WPW#     SPACE  3,14                                             CGTM        118     A
CGTM     **        WPW# - WRITE PSEUDO OP WORD TO *SLIST* FILE             CGTM        119     A
CGTM     *                                                                 CGTM        120     A
CGTM     *         ENTRY  (B3,X6) = OPCODE VALUE & LOW 48 BITS OF WORD     CGTM        121     A
CGTM     *                                                                 CGTM        122     A
CGTM     *         PRESERVES A0, X0, A5, X5                                CGTM        123     A
CGTM                                                                       CGTM        124     A
CGTM      WPW1     WRITEW =XF.SLST,B6,B7                                   CGTM        125     A
CGTM                                                                       CGTM        126     A
CGTM      WPW      ENTRY. **,#                                             CGTM        127     A
CGTM               PX6    B3,X6                                            CGTM        128     A
CGTM               SA6    WPWA                                             CGTM        129     A
CGTM               SB6    A6                                               CGTM        130     A
CGTM               SB7    B1                                               CGTM        131     A
CGTM               EQ     WPW1                                             CGTM        132     A
CGTM                                                                       CGTM        133     A
CGTM      WPWA     BSS    1                                                CGTM        134     A
CGTM      CGEP     TITLE  CGEP - CODE GENERATOR END PROCESSOR ROUTINES     CCG15         1     A
CGTM     **        CGEP - CODE GENERATOR END PROCESSOR                     CCG15         2     A
CGTM     *                                                                 CCG15         3     A
CGTM     *         *CGEP* CONTAINS ROUTINE TO DO FINAL ADDRESS DEFINITION, CCG15         4     A
CGTM     *         BLOCK LENGTH CALCULATION, BLOCK SWITCHING, ETC.         CCG15         5     A
CGTM     *                                                                 CCG15         6         I
          -CCG16
CGTM     *         NOTE THAT SINCE THE CODE IN THE END PROCESSOR IS USED BY THE  CCG15   7         I
          -CCG16
CGTM     *         ASSEMBLER FOR SCRATCH STORAGE IT MAY NOT BE ACCESSED BY IT.   CCG15   8         I
          -CCG16
CGTM                                                                       CCG15         9         I
          -CCG16
CGTM      N$SLBT   ENTRY. 0                                                CCG15        10         I
          -CCG16
CGTM               USE    WII                                              CCG15        11         I
          -CCG16
CGTM                                                                       CCG15        12         I
          -CCG16
CGTM               TABLES (CUT,CVT,PIT)                                    CCG15        13         I
          -CCG16
CGTM                                                                       CCG15        14         I
          -CCG16
CGTM      N$SLBT   ENTRY. 0                                                CCG16        65     A
CGTM      N$IT     ENTRY. 0                                                CCG15        15     A
CGTM      N$OT     ENTRY. 0                                                CCG15        16     A
CGTM                                                                       CCG15        17     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM      CC$LBO   ENTRY. 0          ORDINAL OF LOCAL BLOCK BEING PROCESSED          CCG15      18    A
CGTM      CC$PC    ENTRY. 0          PARCEL COUNTER ( 0 - 3 )                        CCG15      19    A
CGTM      CC$BLEN  ENTRY. 0          LENGTH OF CURRENT BLOCK                         CCG15      20    A
CGTM                                                                                 CCG15      21         I
          -CCG16
CGTM               IFNE   HC.FPAS,0,1                                                CCG15      22         I
          -CCG16
CGTM      LSR      DATA   0          LAST SUB REF  12/P(FPO),30/,18/BLEN             CCG15      23         I
          -CCG16
CGTM                                                                                 CCG15      24    A
CGTM               QUAL   WII                                                        CCG15      25    A
CGTM      CG$RBT   SPACE  2,14                                                       CCG15      26    A
CGTM     **        CG$RBT - RELOCATE BLOCK TABLE                                    CCG15      27    A
CGTM                                                                                 CCG15      28    A
CGTM      CG$RBT   ROUTINE                                                           CCG15      29    A
CGTM               SA3    CC$PC                                                     CCG15      30    A
CGTM               SA4    CC$BLEN                                                   CCG15      31    A
CGTM               SA1    O.LBT                                                     CCG15      32         I
          -CCG16
CGTM               SX1    =XF$LBT                                                   CCG16      66    A
CGTM               SA2    CC$LBO                                                    CCG15      33    A
CGTM               LX3    18                                                        CCG15      34    A
CGTM               BX6    X3+X4                                                     CCG15      35    A
CGTM               SB2    X2                                                        CCG15      36    A
CGTM               SA6    X1+B2       LBT(LBO) = (PC,BLEN)                          CCG15      37    A
CGTM               SA3    L.LBT                                                     CCG15      38         I
          -CCG16
CGTM               SB2    B0          I = 0                                         CCG15      39    A
CGTM               SB3    X3          N = L.LBT                                     CCG15      40         I
          -CCG16
CGTM               SB3    =XZ$LBT+1   N = Z$LBT + 1                                 CCG16      67         I
          -CCGA072
CGTM               SA3    =XN$LBT                                                   CCGA072    39    A
CGTM               SB3    X3+1        N = N$LBT + 1                                 CCGA072    40    A
CGTM               MX0    42                                                        CCG15      41    A
CGTM               MX7    0           S = 0                                         CCG15      42    A
CGTM                                                                                 CCG15      43    A
CGTM      RBT1     SA3    X1+B2                                                     CCG15      44    A
CGTM               BX5    X0*X3                                                     CCG15      45    A
CGTM               ZR     X5,RBT2     IF PC[LBT(I)] = 0                             CCG15      46    A
CGTM               SX3    X3+1        BLEN[LBT(I)] = BLEN[LBT(I)] + 1               CCG15      47    A
CGTM      RBT2     SA7    A3          LBT(I) = S    */ FWA                          CCG15      48    A
CGTM               IX7    X7+X3       S = S + BLEN[LBT(I)]                          CCG15      49    A
CGTM               SB2    B2+1        I = I + 1                                     CCG15      50    A
CGTM               LT     B2,B3,RBT1                                                CCG15      51    A
CGTM                                                                                 CCG15      52    A
CGTM               SA7    N$SLBT                                                    CCG15      53    A
CGTM               EQ     CG$RBT                                                    CCG15      54    A
CGTM      CG$CUB   TITLE  CG$CUB - CHANGE USE BLOCK                                 CCG15      55    A
CGTM     **        CG$CUB - CHANGE USE BLOCK                                        CCG15      56    A
CGTM     *                                                                          CCG15      57    A
CGTM     *         ENTRY  (X1) = *LBT* ORDINAL OF NEW USE BLOCK                     CCG15      58    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM                                                                                    CCG15        59     A
CGTM      CG$CUB   ROUTINE                                                              CCG15        60     A
CGTM               SETB1                                                                CCG15        61     A
CGTM               SA2    CC$LBO      OBO = CC$LBO                                      CCG15        62     A
CGTM               IX7    X1-X2                                                         CCG15        63     A
CGTM               BX6    X1                                                            CCG15        64     A
CGTM               ZR     X7,CG$CUB   IF NBO = OBO                                      CCG15        65     A
CGTM                                                                                    CCG15        66     A
CGTM               SA6    A2          CC$LBO = NBO                                      CCG15        67     A
CGTM               SA3    CC$PC                                                         CCG15        68     A
CGTM               SA4    CC$BLEN                                                       CCG15        69     A
CGTM               LX3    18                                                            CCG15        70          I
          -CCG0220
CGTM               LX3    54                                 HX3   LB.PARC              CCG0220       9     A
CGTM               BX6    X3+X4                                                         CCG15        71     A
CGTM               SA5    O.LBT                                                         CCG15        72          I
          -CCG16
CGTM               SB3    X5                                                            CCG15        73          I
          -CCG16
CGTM               SB3    F$LBT                                                         CCG16        68     A
CGTM               SA6    B3+X2       LBT(OBO) = SHIFT(PC,18) ! BLEN                    CCG15        74     A
CGTM               SA5    B3+X1                                                         CCG15        75     A
CGTM               SX6    X5                                                            CCG15        76     A
CGTM               SA6    A4                                                            CCG15        77     A
CGTM               AX5    18          SET NEW PARCEL AND BLEN                           CCG15        78          I
          -CCG0220
CGTM               AX5    54          PARCNT = PARC[LBTI] / AX5   -LB.PARCL             CCG0220      10     A
CGTM               SX6    X5                                                            CCG15        79     A
CGTM               SA6    A3                                                            CCG15        80     A
CGTM               SX6    X1+1S15                                                       CCG15        81     A
CGTM               WRITEP USE         OUTPUT *USE* TO SLIST                             CCG15        82     A
CGTM               EQ     CG$CUB                                                        CCG15        83     A
CGTM      FSU#     SPACE  3,14                                                          CCG15        84     A
CGTM     **        FSU# - FORCE NEXT SEQUENCE UPPER                                     CCG15        85     A
CGTM                                                                                    CCG15        86     A
CGTM      FSU#     ROUTINE                                                              CCG15        87     A
CGTM               SA1    CC$PC                                                         CCG15        88     A
CGTM               ZR     X1,FSU#     IF PARCEL = 0                                     CCG15        89     A
CGTM               MX7    0                                                             CCG15        90     A
CGTM               SA7    A1          PARCEL = 0                                        CCG15        91     A
CGTM               SA2    CC$BLEN                                                       CCG15        92     A
CGTM               SX6    X2+B1       BLEN = BLEN + 1                                   CCG15        93     A
CGTM               SA6    A2                                                            CCG15        94     A
CGTM               SA6    FU          FU = 1                                            CCG15        95     A
CGTM               EQ     FSU#                                                          CCG15        96     A
CGTM                                                                                    CCG15        97     A
CGTM      FU       DATA   0           "0 IF TO FORCE UPPER                              CCG15        98     A
CGTM      CG$EP    TITLE  CG$EP - CCG END PROCESSOR                                     CCG15        99     A
CGTM     **        CG$EP - CODE GENERATOR END PROCESSOR                                 CCG15       100     A
CGTM     *         FUNCTIONS -                                                          CCG15       101     A
CGTM     *         SETUP *CVT* ORDINAL TABLE FOR *CGIA*                                 CCG15       102     A
CGTM     *         OUPUT TABLE OF CONSTANT LITERALS AS - * CON. BSS 0 FOLLOWED          CCG15       103     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM     *         BY DATA WORDS.                                                  CCG15        104     A
CGTM     *         DEFINE ADDRESS"S AND ALLOCATE STORAGE FOR OTHER SPECIAL SYMBOLS. CCG15        105     A
CGTM     *                                                                         CCG15        106     A
CGTM     *         ENTRY  CVT - CONSTANT VALUE TABLE                               CCG15        107     A
CGTM     *                CUT - CONSTANT USE TABLE ENTRIES ARE 0 OR 1 FOR UNUSED  CCG15        108     A
CGTM     *                   OR USED.                                             CCG15        109     A
CGTM     *                N$IT, N$OT = NUMBER OF IT."S USED, ETC                  CCG15        110     A
CGTM     *                                                                         CCG15        111     A
CGTM     *         EXIT   CVT REFORMATTED AS 60/ ORDINAL OF CONSTANT VALUE IN      CCG15        112     A
CGTM     *                   CON. ARRAY.                                          CCG15        113     A
CGTM     *                ADDRESS OF SPECIAL SYMBOLS ( IT., OT., CON. , ETC )     CCG15        114     A
CGTM     *                   DEFINED AND OUTPUT TO *SLIST* FILE.                  CCG15        115     A
CGTM                                                                               CCG15        116     A
CGTM      CG$EP    ROUTINE                                                         CCG15        117     A
CGTM               SETB1                                                           CCG15        118     A
CGTM                                                                               CCG15        119     A
CGTM     *         REFORMAT *CUT* AS ORDINALS TO CON. ARRAY, SQUEEZE UNUSED        CCG15        120     A
CGTM     *         ENTRIES OUT OF *CVT*.                                           CCG15        121     A
CGTM                                                                               CCG15        122     A
CGTM               ADDWRD CUT,0                                                    CCG15        123     A
CGTM               SA1    O.CVT                                                    CCG15        124           I
          -CCG16
CGTM               SA1    O$CVT                                                    CCG16         69     A
CGTM               SB2    B0          I = 0                                        CCG15        125     A
CGTM               SB3    X3          N = L.CUT                                    CCG15        126     A
CGTM               SB4    X1                      (B4) = CVT                       CCG15        127     A
CGTM               SB5    X2                      (B5) = CUT                       CCG15        128     A
CGTM               SX7    0           J = 0                                        CCG15        129     A
CGTM                                                                               CCG15        130     A
CGTM      EP1      SA5    B5+B2                                                    CCG15        131     A
CGTM               ZR     X5,EP2      IF CUT(I) = 0 */ CON NOT REFERENCED          CCG15        132     A
CGTM                                                                               CCG15        133     A
CGTM               SA7    A5          CUT(I) = J    */ SET ORDINAL                 CCG15        134     A
CGTM               SA4    B4+B2                                                    CCG15        135     A
CGTM               BX6    X4          CVT(J) = CVT(I)  */ MOVE TO FINAL POSITION   CCG15        136     A
CGTM               SA6    B4+X7                                                    CCG15        137     A
CGTM               SX7    X7+B1       J = J + 1                                    CCG15        138     A
CGTM                                                                               CCG15        139     A
CGTM      EP2      SB2    B2+B1       I = I + 1                                    CCG15        140     A
CGTM               LT     B2,B3,EP1   IF I < N                                     CCG15        141     A
CGTM                                                                               CCG15        142     A
CGTM               BX5    X7                                                       CCG15        143     A
CGTM               ZR     X7,EP3      IF J = 0                                     CCG15        144     A
CGTM     *                                                                         CCG15        145     A
CGTM     *         DEFINE ADDRESS OF CON. AND OUTPUT TABLE OF DATA WORDS           CCG15        146     A
CGTM                                                                               CCG15        147     A
CGTM               SA1    =XS$CON                                                  CCG15        148     A
CGTM               SX2    B0                                                       CCG15        149     A
CGTM               RJ     CG$DSA      DEFINE CON.                                  CCG15        150     A
CGTM               SA3    CC$BLEN                                                  CCGA072       41     A
CGTM               IX7    X3+X5                                                    CCGA072       42     A
CGTM               SA7    A3          BLEN = BLEN + L$CVT                          CCGA072       43     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM              BX6    X5                                                  CCG15      151    A
CGTM              LX6    SI.CAP                                              CCG15      152    A
CGTM              WRITEP DATA        OUTPUT DATA HEADER WORD                 CCG15      153    A
CGTM              SA4    O.CVT                                               CCG15      154         I
          -CCG16
CGTM              SA4    O$CVT                                               CCG16       70    A
CGTM              WRITEW F.SLST,X4,X5      OUTPUT DATA WORDS                 CCG15      155    A
CGTM                                                                         CCG15      156    A
CGTM     *        DEFINE ADDRESS"S OF SPECIAL SYMBOLS                         CCG15      157    A
CGTM                                                                         CCG15      158    A
CGTM      EP3     SX6    0                                                   CCG15      159    A
CGTM              SA6    L.CVT       L.CVT = 0                               CCG15      160         I
          -CCG16
CGTM              SA6    L$CVT             L$CVT = 0                         CCG16       71    A
CGTM      .PLI    IFNE   HC.ID,3                                             CCG15      161    A
CGTM              ECHO   4,Z=(IT,OT)                                         CCG15      162    A
CGTM              SA1    =XS$Z                                               CCG15      163    A
CGTM              SA2    N$Z                                                 CCG15      164    A
CGTM     +        ZR     X2,*+1                                              CCG15      165    A
CGTM              RJ     CG$DSA                                              CCG15      166    A
CGTM      .PLI    ENDIF                                                      CCG15      167    A
CGTM                                                                         CCG15      168    A
CGTM              EQ     CG$EP                                               CCG15      169    A
CGTM      CG$DSA  SPACE  3,14                                                CCG15      170    A
CGTM     **       CG$DSA - DEFINE SYMBOL ADDRESS AND OUTPUT STORAGE FOR IT   CCG15      171    A
CGTM     *                                                                   CCG15      172    A
CGTM     *        ENTRY  (X1) = IH, SYMTAB ORDINAL                            CCG15      173    A
CGTM     *               (X2) = WDS , NUMBER OF WORDS RESERVED FOR THIS SYMBOL CCG15    174    A
CGTM     *                                                                   CCG15      175    A
CGTM     *        PRESERVES A0, X0, A5, X5                                    CCG15      176    A
CGTM                                                                         CCG15      177    A
CGTM      CG$DSA  ROUTINE                                                    CCG15      178    A
CGTM              SA4    CC$PC                                               CCG15      179    A
CGTM              SETB1                                                      CCG15      180    A
CGTM              SA3    CC$BLEN                                             CCG15      181    A
CGTM              SX6    B1                                                  CCG15      182    A
CGTM              LX6    WC.RLP                                              CCG15      183    A
CGTM              ZR     X4,DSA1     IF PARCEL = 0                           CCG15      184    A
CGTM              MX7    0                                                   CCG15      185    A
CGTM              SA7    A4          PARCEL = 0                              CCG15      186    A
CGTM              SX3    X3+1        BLEN = BLEN + 1                         CCG15      187    A
CGTM      DSA1    IX7    X3+X2       RA = BLEN;  BLEN = BLEN + WDS           CCG15      188    A
CGTM              SA4    CC$LBO                                              CCG15      189    A
CGTM              SA7    A3                                                  CCG15      190    A
CGTM              ZR     X1,DSA2     IF IH = 0                               CCG15      191    A
CGTM              LX4    WC.RBP                                              CCG15      192    A
CGTM              LX3    WC.RAP                                              CCG16       72    A
CGTM              BX4    X4+X3                                               CCG15      193    A
CGTM              SA3    O.SYM                                               CCG15      194         I
          -CCG16
CGTM              SA3    =XO$SYM                                             CCG16       73    A
CGTM              BX6    X6+X4       ADW = WC(1,0,CG.LBO,RA)                 CCG15      195    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM              SB2    X3+2                                                  CCG15      196    A
CGTM              LX7    B1,X1                                                 CCG15      197    A
CGTM              IX7    X1+X7                                                 CCG15      198    A
CGTM              SA6    B2+X7       SYM(3*IH+2) = ADW                         CCG15      199    A
CGTM      DSA2    LX2    R1.CAP                                                CCG15      200    A
CGTM              BX6    X2+X1                                                 CCG15      201    A
CGTM              WRITEP LAB         OUTPUT * IH BSS WDS * TO SLIST            CCG15      202    A
CGTM              EQ     CG$DSA                                                CCG15      203    A
CGTM      WII     TITLE  WII - WRITE ISSUED INSTRUCTIONS TO *SLIST*            CCG15      204    A
CGTM     **       WII - WRITE ISSUED INSTRUCTIONS IN *SI.* FORMAT TO *SLIST*.  CCG15      205    A
CGTM     *        FUNCTIONS -                                                  CCG15      206    A
CGTM     *                MAINTAIN BLOCK LENGTH AND PARCEL COUNTER             CCG15      207    A
CGTM     *                ADJUST SPECIAL INSTRUCTIONS ( TLD, TST, ETC. )       CCG15      208    A
CGTM     *                SET *MAT* BIT IN WORDB OF *SYM*                      CCG15      209    A
CGTM     *                DEFINE ADDRESS"S OF TRANSFER LABELS                  CCG15      210    A
CGTM                                                                           CCG15      211    A
CGTM      SZ      MICRO  1,, B2      INSTRUCTION SIZE IN PARCELS               CCG15      212    A
CGTM      PC      MICRO  1,, B5      PARCEL - 4                                CCG15      213    A
CGTM      BL      MICRO  1,, B7      BLOCK LENGTH                              CCG15      214    A
CGTM                                                                           CCG15      215    A
CGTM      WII#    ROUTINE                                                       CCG15      216    A
CGTM              TRACE  WII,PIT                                               CCG15      217    A
CGTM              SA5    O.PIT                                                 CCG15      218         I
         -CCG16
CGTM              SA5    O$PIT                                                 CCG16       74    A
CGTM              WRITEW F.SLST,X5,1       OUTPUT *BOS* TO SLIST               CCG15      219    A
CGTM              SA1    FU                                                    CCG15      220    A
CGTM              ZR     X1,WII0     IF FU = 0     */ NO FORCE UPPER            CCG15      221    A
CGTM              MX6    0           FU = 0                                    CCG15      222    A
CGTM              SA6    A1                                                    CCG15      223    A
CGTM              WRITEP BSS         OUTPUT FORCE UPPER (  BSS 0 )             CCG15      224         I
         -CCG30
CGTM              WRITEP LAB                                                   CCG30        1    A
CGTM                                                                           CCG15      225    A
CGTM      WII0    SA5    X5          PI = [O.PIT]                              CCG15      226    A
CGTM              SA2    O.SYM                                                 CCG15      227         I
         -CCG16
CGTM              SA2    O$SYM                                                 CCG16       75    A
CGTM              SB6    X2+B1                    (B6) = SYM + 1               CCG15      228    A
CGTM              SA4    CC$PC                                                 CCG15      229    A
CGTM              SA3    CC$BLEN                                               CCG15      230    A
CGTM              S"BL"  X3                                                    CCG15      231    A
CGTM              S"PC"  X4-4                                                  CCG15      232    A
CGTM              S"SZ"  B0          SZ = 0                                    CCG15      233    A
CGTM              SA2    O.CUT                                                 CCG15      234         I
         -CCG16
CGTM              SA2    O$CUT                                                 CCG16       76    A
CGTM              MX0    -6                                                    CCG15      235    A
CGTM              SA0    X2                       (A0) = CUT                   CCG15      236    A
CGTM                                                                           CCG15      237    A
CGTM     *        RETURN POINT FOR PROCESSING OF MOST INSTRUCTIONS             CCG15      238    A
CGTM                                                                           CCG15      239    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM      WII1     S"PC"   "PC"+"SZ"   PC = PC + SZ                                  CCG15        240     A
CGTM               SA5     A5+B1       PI = PI + 1                                   CCG15        241     A
CGTM               LT      "PC",WII2   IF PC < 4                                     CCG15        242     A
CGTM                                                                                 CCG15        243     A
CGTM               SB3     "PC"        S = PC                                        CCG15        244     A
CGTM               S"PC"   -4          PC = 0                                        CCG15        245     A
CGTM               S"BL"   "BL"+B1     BLEN = BLEN + 1                               CCG15        246     A
CGTM               ZR      B3,WII2     IF S = 0      */ NO SIZE FAULT                CCG15        247     A
CGTM                                                                                 CCG15        248     A
CGTM               S"PC"   "PC"+"SZ"   PC = SZ                                       CCG15        249     A
CGTM                                                                                 CCG15        250     A
CGTM     *         SETUP REGISTERS, JUMP TO INSTRUCTION PROCESSOR                    CCG15        251     A
CGTM                                                                                 CCG15        252     A
CGTM      WII2     UX7     B3,X5                                                     CCG15        253     A
CGTM               SA3     OC.AT+B3                                                  CCG15        254         I
          -CCG16
CGTM               S"SZ"   B1          SZ = 1                                        CCG15        255         I
          -CCG16
CGTM               AX3     18                                                        CCG15        256         I
          -CCG16
CGTM               SX6     B3                                                        CCG16         77     A
CGTM               S"SZ"   B1          SZ = 1                                        CCG16         78     A
CGTM               LX6     -1                                                        CCG16         79     A
CGTM               SA3     WIIA+X6                                                   CCG16         80     A
CGTM               SX4     30                                                        CCG16         81     A
CGTM               AX6     59                                                        CCG16         82     A
CGTM               BX4     -X6*X4      0 OR 30                                       CCG16         83     A
CGTM               SB4     X4                                                        CCG16         84     A
CGTM               LX3     B4                                                        CCG16         85     A
CGTM               SB4     X3                                                        CCG15        257     A
CGTM               JP      B4+WII1     JUMP TO PROCESSOR                             CCG15        258     A
CGTM       NOP     SPACE   2,14                                                      CCG15        259     A
CGTM     **        NOP - TERMINATE PROCESSING                                        CCG15        260     A
CGTM                                                                                 CCG15        261     A
CGTM               PROCESS NOP                                                       CCG15        262     A
CGTM               SX6     "PC"+4                                                    CCG15        263     A
CGTM               SA6     CC$PC                                                     CCG15        264     A
CGTM               SX7     "BL"+                                                     CCG15        265     A
CGTM               SA7     CC$BLEN                                                   CCG15        266     A
CGTM               SA4     O.PIT                                                     CCG15        267         I
          -CCG16
CGTM               SA3     L.PIT                                                     CCG15        268         I
          -CCG16
CGTM               SA4     =XO$PIT                                                   CCG16         86     A
CGTM               SA3     =XL$PIT                                                   CCG16         87     A
CGTM               MX7     0                                                         CCG15        269     A
CGTM               SA7     A3          L.PIT = 0                                     CCG15        270     A
CGTM               WRITEW  F.SLST,X4+B1,X3-2 WRITE REST OF *PIT* TO SLIST            CCG15        271     A
CGTM               EQ      WII#                                                      CCG15        272     A
CGTM       DRL     SPACE   2                                                         CCG16         88     A
CGTM       .FPAS   IFNE    HC.FPAS*.DAL      F.P. ADDSUB & LEVEL 2                   CCG16         89     A
CGTM                                                                                 CCG16         90     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM                PROCESS (DRL,DWL)                                                   CCG16        91     A
CGTM                AX7    SI.IHP                                                       CCG16        92     A
CGTM                SB4    X7                                                           CCG16        93     A
CGTM                ZR     X4,WII1     IF IH[PI] = 0 */ NOT A *SUB0*                    CCG16        94          I
          -CCG35
CGTM                                                                                    CCG16        95          I
          -CCG35
CGTM                LX6    B1,X7                                                        CCG16        96          I
          -CCG35
CGTM                ZR     B4,WII1     IF IH[PI] = 0 */ NOT A *SUB0*            001240  CCG35         4     A
CGTM                SB4    X7-IH.LCM                                                001250  CCG35         5     A
CGTM                SX6    B4+B4                                                    001260  CCG35         6     A
CGTM                SB4    B6+B4                                                            CCG16        97     A
CGTM                SA4    B4+X6       WORDB = SYM(3*IH+1)                              CCG16        98     A
CGTM                MX7    -WB.FPOL                                                         CCG16        99     A
CGTM                LX4    -WB.FPOP                                                         CCG16       100     A
CGTM                SA3    =XO$FPI                                                          CCG16       101     A
CGTM                BX6    -X7*X4                                                           CCG16       102     A
CGTM                SB4    X3                                                               CCG16       103          I
          -CCG24
CGTM                SB4    X3-1                                                             CCG24         1     A
CGTM                SA2    B4+X6                                                            CCG16       104     A
CGTM                SX3    B1                                                               CCG16       105     A
CGTM                LX3    FP.SUB0P                                                         CCG16       106     A
CGTM                IX6    X2+X3       SUB0[FPI(FPO)] = SUB0[FPI(FPO)] + 1              CCG16       107     A
CGTM                SA6    =XCC$SUB0   CC$SUB0 = 1                                      CCG18        59     A
CGTM                SA6    A2                                                           CCG16       108     A
CGTM                EQ     WII1                                                         CCG16       109     A
CGTM      .FPAS     ENDIF                                                                CCG16       110     A
CGTM      TLD       SPACE  3,14                                                         CCG15       273     A
CGTM     **         TLD, TST - ADJUST *CA* TO REFLECT ASSIGNMENT MADE BY *GPO*           CCG15       274     A
CGTM                                                                                    CCG15       275     A
CGTM      TLD       PROCESS (TLD,TST)                                                    CCG15       276     A
CGTM                LX7    -SI.RJP                                                      CCG15       277     A
CGTM                BX6    -X0*X7                                                       CCG15       278     A
CGTM                NZ     X6,WII1     IF RJ[PI] " 0 */ CONVERTED TO A SHORT INSTRUCTION CCG15       279     A
CGTM                                                                                    CCG15       280     A
CGTM     *          SEARCH *TET* TO FIND LOCATION IN IT. BLOCK ASSIGNED                  CCG15       281     A
CGTM     *          TO THIS TLD/TST .                                                    CCG15       282     A
CGTM                                                                                    CCG15       283     A
CGTM                SA1    =XO$TET                                                      CCG15       284     A
CGTM                SB3    59-T.FAP                                                     CCG15       285     A
CGTM                SB2    X1                          (B2) = TET                       CCG15       286     A
CGTM                LX7    SI.RJP-SI.CAP                                                CCG15       287     A
CGTM                SB4    59-T.EQVP                                                    CCG15       288     A
CGTM                                                                                    CCG15       289     A
CGTM      TLD1      SA1    B2+X7       TI = TET(C)                                      CCG15       290     A
CGTM                LX6    B3,X1                                                        CCG15       291     A
CGTM                SX7    X1          C = CA[TI]                                       CCG15       292     A
CGTM                LX2    B4,X1                                                        CCG15       293     A
CGTM                MI     X6,TLD3     IF FA[TI]     */ FINAL ASSIGNMENT MADE           CCG15       294     A
CGTM                MI     X2,TLD1     IF EQV[TI]    */ EQUIVALENT TO ANOTHER           CCG15       295     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM                                                                                    CCG15        296     A
CGTM     *         CA NOT ASSIGNED, MUST BE OPT=1, GET NEXT CELL AND SAVE               CCG15        297     A
CGTM     *         VALUE IN *CA* FIELD OF *TET* ENTRY.                                  CCG15        298     A
CGTM                                                                                    CCG15        299     A
CGTM               SA2    =XN.GT#                                                       CCG15        300     A
CGTM               BX7    X2          C = N.GT                                          CCG15        301     A
CGTM               MX3    -T.CAL                                                        CCG15        302     A
CGTM               SX6    X2+B1       N.GT = N.GT + 1                                   CCG15        303     A
CGTM               SA6    A2                                                            CCG15        304     A
CGTM               SA4    =XN$IT                                                        CCG15        305     A
CGTM               SX2    B1                                                            CCG15        306     A
CGTM               BX1    X3*X1                                                         CCG15        307     A
CGTM               BX1    X1+X7       CA[TI] = C                                        CCG15        308     A
CGTM               LX2    T.FAP                                                         CCG15        309     A
CGTM               BX6    X1+X2       FA[TI] = 1                                        CCG15        310     A
CGTM               SA6    A1                                                            CCG15        311     A
CGTM               SX6    X7+B1                                                         CCG15        312     A
CGTM               IX4    X4-X6                                                         CCG15        313     A
CGTM               PL     X4,TLD3     N.IT = MAX( N.IT , C+1 )                          CCG15        314     A
CGTM               SA6    A4                                                            CCG15        315     A
CGTM                                                                                    CCG15        316     A
CGTM      TLD3     LX5    -SI.CAP                                                       CCG15        317     A
CGTM               MX1    -SI.CAL                                                       CCG15        318     A
CGTM               BX5    X1*X5                                                         CCG15        319     A
CGTM               BX7    X5+X7       CA[PI] = C                                        CCG15        320     A
CGTM               LX7    SI.CAP                                                        CCG15        321     A
CGTM               S"SZ"  2           SZ = 2                                            CCG15        322     A
CGTM               SA7    A5                                                            CCG15        323     A
CGTM               EQ     WII1                                                          CCG15        324     A
CGTM      PLD      SPACE  2,12                                                          CCG15        325     A
CGTM               PROCESS (PLD,PST,S)                                                  CCG15        326     A
CGTM               AX7    SI.CAP                                                        CCG15        327     A
CGTM               ZR     X7,WII1     IF CA[PI] = 0                                     CCG15        328     A
CGTM               S"SZ"  B1+B1       SZ = 2                                            CCG15        329     A
CGTM               EQ     WII1                                                          CCG15        330     A
CGTM      LD       SPACE  3,14                                                          CCG15        331     A
CGTM               PROCESS ILD                                                          CCG15        332     A
CGTM               SA2    =XS$IT                                                        CCG15        333     A
CGTM               LX5    -SI.IHP                                                       CCG15        334     A
CGTM               BX1    X2-X5                                                         CCG15        335     A
CGTM               SB4    X1                                                            CCG15        336     A
CGTM               LX5    SI.IHP                                                        CCG15        337     A
CGTM               ZR     X1,TLD      IF IH[PI] = IT.                                   CCG15        338         I
          -CCGA028
CGTM               ZR     B4,TLD      IF IH[PI] = IT.
>>>>      CCGA028 11                                                                    CCGA028        5     A
CGTM                                                                                    CCG15        339     A
CGTM               PROCESS (LD,ST,STT)                                                  CCG15        340     A
CGTM               AX7    SI.CAIHP                                                      CCG15        341     A
CGTM               ZR     X7,WII1     IF CAIH[PI] = 0                                   CCG15        342     A
CGTM               S"SZ"  B1+B1       SZ = 2                                            CCG15        343     A
CGTM                                                                                    CCG15        344     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM              MX1    -IH.IL                                          CCG15        345     A
CGTM              LX1    IH.IP                                           CCG15        346     A
CGTM              BX2    -X1*X7                                          CCG15        347     A
CGTM              NZ     X2,LD1      IF I[PI] " 0  */ NOT IN SYM         CCG15        348     A
CGTM              IF     -DEF,WB.MATP,1                                  CCG15        349     A
CGTM              IFNE   HC.FPAS,0,3                                     CCG15        350     A
CGTM              SX7    X7
>>>>     CCGA037 13                                                      CCGA037        7     A
CGTM              LX6    B1,X7                                           CCG15        351     A
CGTM              SB3    X7+B6                                           CCG15        352     A
CGTM              SA3    B3+X6       WORDB = SYM(3*IH+1)                 CCG15        353     A
CGTM                                                                    CCG15        354     A
CGTM       .MAT   IF     DEF,WB.MATP                                     CCG15        355     A
CGTM              SX1    B1                                              CCG15        356     A
CGTM              LX1    WB.MATP                                         CCG15        357     A
CGTM              BX6    X1+X3       MAT[SYM(3*IH+1)] = 1                CCG15        358     A
CGTM              SA6    A3                                              CCG15        359     A
CGTM       .MAT   ENDIF                                                  CCG15        360     A
CGTM                                                                    CCG15        361     A
CGTM       .FPAS  IFNE   HC.FPAS,0                                       CCG15        362     A
CGTM              LX3    59-WB.FPP                                       CCG15        363     A
CGTM     +        PL     X3,*+1      IF ^FP[WORDB]                       CCG15        364     A
CGTM              RJ     CSR         COUNT SUB REF                       CCG15        365     A
CGTM       .FPAS  ENDIF                                                  CCG15        366     A
CGTM                                                                    CCG15        367     A
CGTM       LD1    LX5    59-SI.H2P                                       CCG15        368     A
CGTM              PL     X5,WII1     IF ^HI[PI]                          CCG15        369     A
CGTM              SA5    A5+1        PI = PI + 1   */ SKIP WORD WITH *H2* IN IT  CCG15  370     A
CGTM              EQ     WII1                                            CCG15        371     A
CGTM       LDC    SPACE  2                                               CCG15        372     A
CGTM              PROCESS LDC                                            CCG15        373     A
CGTM              LX5    -SI.CAP                                         CCG15        374     A
CGTM              SX6    B1                                              CCG15        375     A
CGTM              S"SZ"  B1+B1       SZ = 2                              CCG15        376     A
CGTM              SB3    A0                                              CCG15        377     A
CGTM              SA6    B3+X5       CUT(CA[SI]) = 1                     CCG15        378     A
CGTM              EQ     WII1                                            CCG15        379     A
CGTM       LDV    SPACE  2,14                                            CCG16        111     A
CGTM       .VD    IF     DEF,VD.MATP                                     CCG16        112     A
CGTM                                                                    CCG16        113     A
CGTM     *        LDV - CONVERT CA TO FINAL CA AND CHANGE OPCODE TO A *LD*  CCG16     114     A
CGTM                                                                    CCG16        115     A
CGTM              PROCESS LDV                                            CCG16        116     A
CGTM              LX7    -SI.CAP                                         CCG16        117     A
CGTM              SX1    X7                                              CCG16        118     A
CGTM              RJ     CG$AVO      CA = CG$AVO( CA[PI] )               CCG16        119     A
CGTM              LX5    -SI.CAP                                         CCG16        120     A
CGTM              MX7    -SI.CAL                                         CCG16        121     A
CGTM              SB3    OC.LD                                           CCG16        122     A
CGTM              BX5    X7*X5                                           CCG16        123     A
CGTM              BX6    X5+X1                                           CCG16        124     A
CGTM              LX6    SI.CAP                                          CCG16        125     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM              PX7    B3,X6       OC[PI] = OC.LD                          CCG16    126    A
CGTM              SA7    A5                                                  CCG16    127    A
CGTM              S"SZ"  B1+B1       SZ = 2                                  CCG16    128    A
CGTM              EQ     WII1                                                CCG16    129    A
CGTM      .VD     ENDIF                                                      CCG16    130    A
CGTM      JPX     SPACE  3                                                   CCG15    380    A
CGTM     *        JUMPS                                                      CCG15    381    A
CGTM                                                                         CCG15    382    A
CGTM              PROCESS (JPX,JPBB)                                         CCG15    383    A
CGTM              SB4    SI.IHP                                              CCG15    384    A
CGTM              RJ     RLV         CHECK REFERENCED LABEL                  CCG15    385    A
CGTM              S"SZ"  2           SZ = 2                                  CCG15    386    A
CGTM              EQ     WII1                                                CCG15    387    A
CGTM                                                                         CCG15    388    A
CGTM              PROCESS JIN                                                CCG15    389    A
CGTM              SB4    SI.IHP                                              CCG15    390    A
CGTM              RJ     RLV         CHECK REFERENCED LABEL                  CCG15    391    A
CGTM              EQ     UJP1                                                CCG15    392    A
CGTM                                                                         CCG15    393    A
CGTM              PROCESS UJP                                                CCG15    394    A
CGTM              SB4    B0                                                  CCG15    395    A
CGTM              RJ     RLV         CHECK REFERENCED LABEL                  CCG15    396    A
CGTM                                                                         CCG15    397    A
CGTM     *        UNCONDITIONAL JUMP, FORCE UPPER AFTER                      CCG15    398    A
CGTM                                                                         CCG15    399    A
CGTM              PROCESS RJ3                                                CCG15    400    A
CGTM      UJP1    SX6    "PC"+B1     S = PC + 1                              CCG15    401    A
CGTM              S"BL"  "BL"+B1     BLEN = BLEN + 1                         CCG15    402    A
CGTM              S"PC"  -4          PC = 0                                  CCG15    403    A
CGTM              SA5    A5+B1       PI = PI + 1                             CCG15    404    A
CGTM              MI     X6,WII2     IF S < 3      */ OLD(PC)+1 < 3          CCG15    405    A
CGTM              S"BL"  "BL"+B1     BLEN = BLEN + 1                         CCG15    406    A
CGTM              EQ     WII2                                                CCG15    407    A
CGTM      RJXJ    SPACE  2,8                                                 CCG15    408    A
CGTM      .RJXJ   IFNE   HC.RJXJ,0                                           CCG15    409    A
CGTM                                                                         CCG15    410    A
CGTM              PROCESS RJXJ                                               CCG15    411    A
CGTM              SX6    "PC"+4                                              CCG15    412    A
CGTM              ZR     X6,RJXJ1    IF PC = 0                               CCG15    413    A
CGTM              S"PC"  -4          PC = 0        */ FORCE UPPER BEFORE     CCG15    414    A
CGTM              S"BL"  "BL"+1      BLEN = BLEN + 1                         CCG15    415    A
CGTM      RJXJ1   SA5    A5+B1       PI = PI + 1                             CCG15    416    A
CGTM              S"BL"  "BL"+B1     BLEN = BLEN + 1                         CCG15    417    A
CGTM              EQ     WII2                                                CCG15    418    A
CGTM                                                                         CCG15    419    A
CGTM      .RJXJ   ENDIF                                                      CCG15    420    A
CGTM      RJ6     SPACE  3,14                                                CCG15    421    A
CGTM      .RJ6    IFNE   HC.RJ6,0                                            CCG15    422    A
CGTM                                                                         CCG15    423    A
CGTM              PROCESS RJ6                                                CCG15    424    A
CGTM              SX6    "PC"+4                                              CCG15    425    A
CGTM              ZR     X6,RJ61     IF PC = 0                               CCG15    426    A
```

```
         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM              S"PC"  -4          PC = 0                                  CCG15      427    A
CGTM              S"BL"  "BL"+1      BLEN = BLEN + 1                         CCG15      428    A
CGTM      RJ61    BSS    0                                                  CCG15      429    A
CGTM      .FPAS   IFNE   HC.FPAS,0                                          CCG15      430    A
CGTM              SB3    X5+B6                                               CCG15      431    A
CGTM              LX6    B1,X5                                               CCG15      432    A
CGTM              SA3    B3+X6                                               CCG15      433    A
CGTM              LX3    59-WB.FPP                                           CCG15      434    A
CGTM     +        PL     X3,*+1      IF ^FP[WORDB]                           CCG15      435    A
CGTM              RJ     CSR         COUNT SUB REF                           CCG15      436    A
CGTM      .FPAS   ENDIF                                                     CCG15      437    A
CGTM                                                                        CCG15      438    A
CGTM              LX5    -SI.CAP                                             CCG15      439    A
CGTM              SX1    X5-7777B                                            CCG15      440         I
          -CCGA041
CGTM              MI     X1,RJ62     IF CA[R1] < 7777B                       CCG15      441         I
          -CCGA041
CGTM              SX1    X5-7776B                                            CCGA041      6    A
CGTM              MI     X1,RJ62     IF CA[R1] < 4095                        CCGA041      7    A
CGTM              S"BL"  "BL"+1      BLEN = BLEN + 1                         CCG15      442    A
CGTM      RJ62    SA5    A5+B1       PI = PI + 1                             CCG15      443    A
CGTM              S"BL"  "BL"+B1     BLEN = BLEN + 1  */ COUNT RJ WORD       CCG15      444    A
CGTM              EQ     WII2                                                CCG15      445    A
CGTM                                                                        CCG15      446    A
CGTM      .RJ6    ENDIF                                                     CCG15      447    A
CGTM      CSR     SPACE  3,14                                              CCG15      448    A
CGTM      .FPAS   IFNE   HC.FPAS,0                                          CCG15      449    A
CGTM     **       CSR - COUNT *SUB* REFERENCE                              CCG15      450    A
CGTM     *                                                                  CCG15      451    A
CGTM     *        ENTRY  (A3) = ADDRESS OF WORD B                           CCG15      452    A
CGTM     *               (X3) = SHIFT(WORDB,59-WB.FPP)                      CCG15      453    A
CGTM     *               (X5) = [PI]                                        CCG15      454    A
CGTM                                                                        CCG15      455    A
CGTM      CSR     ROUTINE                                                   CCG15      456    A
CGTM              SA1    O.FPT                                               CCG15      457         I
          -CCG16
CGTM              SA1    =XO$FPI                                             CCG16      131    A
CGTM              LX3    1+WB.FPP-WB.FPOP                                    CCG15      458    A
CGTM              MX4    -WB.FPOL                                            CCG15      459    A
CGTM              BX2    -X4*X3                                              CCG15      460    A
CGTM              SB3    X2                                                  CCG15      461         I
          -CCG18
CGTM              SB3    X2-1                                                CCG18       60    A
CGTM              SA2    B3+X1                                               CCG15      462    A
CGTM              SX4    B1                                                  CCG15      463    A
CGTM              LX4    FP.LENP                                             CCG18       61    A
CGTM              IX6    X2+X4       FPIT(FPO) = FPIT(FPO) + 1               CCG15      464         I
          -CCG16
CGTM              IX6    X2+X4       FPI(FPO) = FPI(FPO) + 1                 CCG16      132    A
CGTM              SA6    A2                                                  CCG15      465    A
CGTM              SA6    =XCC$SUB    CC$SUB = 1    */ INDICATE SUBS IN PROGRAM  CCG18       62    A
CGTM              SA1    LSR         OLSR = LSR                              CCG15      466         I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

          -CCG16
CGTM                SX7    "BL"                                                  CCG15        467          I
          -CCG16
CGTM                PX7    B3,X7                                                 CCG15        468          I
          -CCG16
CGTM                SA7    A1          LSR = PACK(FPO,BLEN)                      CCG15        469          I
          -CCG16
CGTM                BX2    X1-X7                                                 CCG15        470          I
          -CCG16
CGTM                NZ     X2,CSR      IF OLSR " LSR */ NOT REF TO SAME FP IN A WD  CCG15     471          I
          -CCG16
CGTM                                                                             CCG15        472          I
          -CCG16
CGTM                IX6    X6+X4       FPIT(FPO) = FPIT(FPO) + 1 */ COUNT *DELAY*  CCG15      473          I
          -CCG16
CGTM                SA6    A6                                                    CCG15        474          I
          -CCG16
CGTM                EQ     CSR                                                   CCG15        475     A
CGTM       .FPAS    ENDIF                                                        CCG15        476     A
CGTM       LAB      SPACE  3,14                                                  CCG15        477     A
CGTM      **        LAB    CA,IH - DEFINE LABEL ADDRESS                          CCG15        478     A
CGTM                                                                             CCG15        479     A
CGTM                PROCESS ENT                                                  CCG15        480     A
CGTM                SX4    B1                                                    CCG15        481     A
CGTM                LX4    SI.CAP                                                CCG15        482     A
CGTM                BX5    X4+X5       CA = 1                                    CCG15        483     A
CGTM                                                                             CCG15        484     A
CGTM                PROCESS LAB                                                  CCG15        485     A
CGTM                SX6    "PC"+4                                                CCG15        486     A
CGTM                ZR     X6,LAB1     IF PC = 0                                 CCG15        487     A
CGTM                S"PC"  -4          PC = 0                                    CCG15        488     A
CGTM                S"BL"  "BL"+1      BLEN = BLEN + 1                           CCG15        489     A
CGTM                                                                             CCG15        490     A
CGTM       LAB1     SA1    CC$LBO                                                CCG15        491     A
CGTM                SX2    B1                                                    CCG15        492     A
CGTM                LX2    WC.RLP                                                CCG15        493     A
CGTM                SX3    "BL"                                                  CCG15        494     A
CGTM                LX3    WC.RAP                                                CCG16        133     A
CGTM                LX1    WC.RBP                                                CCG15        495     A
CGTM                BX2    X1+X2                                                 CCG15        496     A
CGTM                BX6    X2+X3       ADW = WC(1,0,CG.LBO,BLEN)                 CCG15        497     A
CGTM                AX7    IH.IP                                                 CCG15        498     A
CGTM                MX1    -IH.IL                                                CCG15        499     A
CGTM                BX7    -X1*X7                                                CCG15        500     A
CGTM                SA1    LABA        MASK OF ADDRESS FIELDS                    CCG38          5     A
CGTM                ZR     X7,LAB2     IF I[PI] = 0                              CCG15        501     A
CGTM                                                                             CCG15        502     A
CGTM                SA3    O.GLT-1+X7  BASE                                      CCG15        503          I
          -CCG16
CGTM                SA3    =XF$FRT-1+X7      TABLE POINTER ADDRESS                CCG16        134     A
CGTM                SA3    X3          BASE                                      CCG16        135     A
CGTM                MX4    -IH.HL                                                CCG15        504     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM              SB4    X7                                                  CCGA021      8    A
CGTM              BX7    -X4*X5                                              CCG15      505    A
CGTM              SB3    X7                                                  CCG15      506    A
CGTM              SA6    X3+B3       TBL(H) = ADW                            CCG15      507         I
          -CCG38
CGTM              SA2    X3+B3                                               CCG38        6    A
CGTM              BX2    -X1*X2      MASK OUT ADDRESS FIELDS                 CCG38        7    A
CGTM              BX6    X2+X6       BUILD NEW C WORD                        CCG38        8    A
CGTM              SA6    A2                                                  CCG38        9    A
CGTM              EQ     LAB3                                                CCG15      508         I
          -CCGA021
CGTM              NE     B4,B1,LAB3  IF I[PI] " 1  */ NOT A GL               CCGA021      9    A
CGTM              SA2    =XL.ELT#                                            CCGA021     10    A
CGTM              ZR     X2,LAB3     IF L.ELT = 0  */ NO ELIMINATED LABEL DEFS CCGA021   11    A
CGTM                                                                         CCGA021     12    A
CGTM     *        SEARCH ELT TO SEE IF THIS LABEL IS EQUIVALENT TO A          CCGA021     13    A
CGTM     *        PROGRAMMER DEFINED LABEL, AND DEFINE THE ADDRESS OF IT.     CCGA021     14    A
CGTM                                                                         CCGA021     15    A
CGTM              SA3    =XO.ELT#                                            CCGA021     16    A
CGTM              SA7    X3          [ELT] = H     */ STORE SEARCH TERMINATOR CCGA021     17    A
CGTM              SB4    X2                                                  CCGA021     18    A
CGTM              SA3    X3+B4       SEARCH TABLE                            CCGA021     19    A
CGTM     +        BX4    X3-X7                                               CCGA021     20    A
CGTM              SA3    A3-B1                                               CCGA021     21    A
CGTM              SB4    X4                                                  CCGA021     22    A
CGTM              AX4    18                                                  CCGA021     23    A
CGTM              NZ     B4,*-1                                              CCGA021     24    A
CGTM              ZR     X4,LAB3     IF NO EQUIVALENT                        CCGA021     25    A
CGTM                                                                         CCGA021     26    A
CGTM              BX5    X4          SET IH OF EQUIVALENT, CA MUST BE 0      CCGA021     27    A
CGTM                                                                         CCG15      509    A
CGTM      LAB2    LX7    B1,X5                                               CCG15      510    A
CGTM              SB3    B6+X5                                               CCG15      511    A
CGTM              SX7    X7+B1                                               CCG15      512    A
CGTM              SA6    B3+X7       SYM(3*IH+2) = ADW                       CCG15      513         I
          -CCG38
CGTM              SA2    B3+X7       RETRIEVE WORDC                          CCG38       10    A
CGTM              BX2    -X1*X2      BUILD NEW WORDC USING LABA MASK         CCG38       11    A
CGTM              BX6    X2+X6                                               CCG38       12    A
CGTM              SA6    A2          SYM(3*IH+2) = ADW                       CCG38       13    A
CGTM                                                                         CCG15      514    A
CGTM      LAB3    LX5    -SI.CAP                                             CCG15      515    A
CGTM              S"BL"  "BL"+X5     BLEN = BLEN + CA[PI]                    CCG15      516    A
CGTM              SA5    A5+1        PI = PI + 1                             CCG15      517    A
CGTM              EQ     WII2                                                CCG15      518    A
CGTM                                                                         CCG38       14    A
CGTM      LABA    BFMW   WC,(RL,RB,RA)                                       CCG38       15    A
CGTM      RLV     SPACE  3,14                                                CCG15      519    A
CGTM     **       RLV - CHECK REFERENCED LABEL VALUE FOR SUBSTITUTION        CCG15      520    A
CGTM     *                                                                   CCG15      521    A
CGTM     *        ENTRY  (B4) = SHIFT COUNT TO FIND *IH* IN *PI* WORD        CCG15      522    A
CGTM                                                                         CCG15      523    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM      RLV      ROUTINE                                                    CCG15      524     A
CGTM               AX6     B4,X5                                              CCG15      525     A
CGTM               MX3     -IH.IL                                             CCG15      526     A
CGTM               SB3     B6+X6                                              CCG15      527     A
CGTM               LX3     IH.IP                                              CCG15      528     A
CGTM               BX7     -X3*X6                                             CCG15      529     A
CGTM               NZ      X7,RLV      IF I[PI] " 0  */ NOT IN *SYM*          CCG15      530     A
CGTM               SX6     X6                                                 CCG15      531     A
CGTM               LX7     B1,X6                                              CCG15      532     A
CGTM               SA3     B3+X7       WORDB = SYM(3*H+1)                     CCG15      533     A
CGTM               LX3     59-WB.LCP                                          CCG15      534     A
CGTM               PL      X3,RLV      IF ^LC[WORDB]                          CCG15      535     A
CGTM                                                                          CCG15      536     A
CGTM               SA2     A3-B1       WORDA = WORDB - 1                      CCG15      537     A
CGTM               MX1     -WA.LCOL                                           CCG15      538     A
CGTM               LX2     -WA.LCOP                                           CCG15      539     A
CGTM               BX3     -X1*X2      L = LCO[WORDA]                         CCG15      540     A
CGTM               SA1     =XO$LCT                                            CCG15      541         I
          -CCG18
CGTM               SA1     =XO.LCT#                                           CCG18       63     A
CGTM               SA2     =XO$BIT                                            CCG15      542     A
CGTM               SB2     X1                                                 CCG15      543     A
CGTM               SB3     X2                                                 CCG15      544     A
CGTM                                                                          CCG15      545     A
CGTM      RLV1     SA1     B2+X3       LCW = LCT(L)                           CCG15      546     A
CGTM               SA3     =XBN#                                              CCG15      547     A
CGTM               MX5     -BI.OLNL                                           CCG15      548     A
CGTM                                                                          CCG15      549     A
CGTM      RLV2     BX7     X1-X3                                              CCG15      550     A
CGTM               BX7     -X5*X7                                             CCG15      551     A
CGTM               ZR      X7,RLV3     IF K = HBN[LCW]                        CCG15      552     A
CGTM               SA3     B3+X3                                              CCG15      553     A
CGTM               LX3     -BI.OLNP                                           CCG15      554     A
CGTM               BX3     -X5*X3      K = OLN[BIT(K)]                        CCG15      555     A
CGTM               NZ      X3,RLV2     IF K " 0                               CCG15      556     A
CGTM                                                                          CCG15      557     A
CGTM               LX1     -LC.NEXTP                                          CCG15      558     A
CGTM               SX3     X1          L = NEXT[LCW]                          CCG15      559     A
CGTM               NZ      X3,RLV1     IF L " 0      */ NOT END OF CHAIN      CCG15      560     A
CGTM               EQ      RLV                                                CCG15      561     A
CGTM                                                                          CCG15      562     A
CGTM      RLV3     LX1     -LC.GLNP                                           CCG15      563     A
CGTM               SX7     X1+I.GL     IH = GLN[LCW] + I.GL                   CCG15      564     A
CGTM               LX6     B4                                                 CCG15      565     A
CGTM               SA5     A5                                                 CCG38       16     A
CGTM               BX5     X5-X6                                              CCG15      566     A
CGTM               LX7     B4                                                 CCG15      567     A
CGTM               BX7     X5+X7       SUBSTITUTE NEW VALUE                   CCG15      568     A
CGTM               SA7     A5                                                 CCG15      569     A
CGTM               EQ      RLV                                                CCG15      570     A
CGTM                                                                          CCG16      136     A
CGTM               MACRO   OPR,N                                              CCG16      137     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM              IFLT   OC.N,OC.CLR,4                                    CCG16     138    A
CGTM              IF     DEF,/WII/.N,2                                    CCG16     139    A
CGTM              VFD    12/,18//WII/.N-/WII/WII1                         CCG16     140    A
CGTM              SKIP   1                                                CCG16     141    A
CGTM              VFD    30/0                     N                       CCG16     142    A
CGTM              ENDM                                                    CCG16     143    A
CGTM                                                                      CCG16     144    A
CGTM       WIIA   BSS    0                                                CCG16     145    A
CGTM      *CALL   OPRDEFS                                                 CCG16     146    A
CGTM              BSS    0                                                CCG16     147    A
CGTM                                                                      CCG15     571    A
CGTM              QUAL   *                                                CCG15     572    A
CGTM       .Z     SET    *-L.WII     AVAILABLE SPACE                      CCG15     573         I
          -CCG16
CGTM              IFLT   .Z,0,1                                           CCG15     574         I
          -CCG16
CGTM              BSS    -.Z                                              CCG15     575         I
          -CCG16
CGTM              USE    *                                                CCG15     576         I
          -CCG16
CGTM                                                                      CCG15     577    A
CGTM              ECHO   2,Z=(CG$RBT,CG$CUB,FSU#,CG$DSA,CG$EP,WII#)       CCG15     578    A
CGTM       Z      EQU    /WII/Z                                           CCG15     579    A
CGTM              ENTRY  Z                                                CCG15     580    A
CGTM       AVO    SPACE  3,14                                             CCG15     581    A
CGTM       .VD    IF     DEF,VD.MATP                                      CCG15     582    A
CGTM      **      CG$AVO - ADJUST VARDIM ORDINAL OF REF TO VD. CELL       CCG15     583    A
CGTM      *                                                               CCG15     584    A
CGTM      *       ENTRY  (X1) = C, INDEX TO VDI                           CCG15     585    A
CGTM      *       EXIT   (X1) = CA, INDEX TO VD. CELL                     CCG15     586    A
CGTM      *       USES   X - 3, 4, 6  A - 1, 3, 4, 6                      CCG15     587    A
CGTM                                                                      CCG15     588    A
CGTM       AVO1   LX3    1+VD.MATP-VD.CAP                                 CCG15     589    A
CGTM              SX1    X3          CA = CA[VE]                          CCG15     590    A
CGTM       CG$AVO ENTRY. **                                              CCG15     591    A
CGTM              SA4    =XO$VDI                                          CCG15     592    A
CGTM              IX6    X4+X1                                            CCG15     593    A
CGTM              SA3    X6          VE = VDI(C)                          CCG15     594    A
CGTM              LX3    59-VD.MATP                                       CCG15     595    A
CGTM              MI     X3,AVO1     IF MAT[VE]    */ CA ASSIGNED         CCG15     596    A
CGTM                                                                      CCG15     597    A
CGTM              MX4    1                                                CCG15     598    A
CGTM              BX3    X3+X4       MAT[VE] = 1   */  SET MAT BIT        CCG15     599    A
CGTM              SA1    =XN$VD      CA = N$VD                            CCG15     600    A
CGTM              LX3    1+VD.MATP-VD.CAP                                 CCG15     601    A
CGTM              MX4    -VD.CAL                                          CCG15     602    A
CGTM              BX3    X4*X3                                            CCG15     603    A
CGTM              BX6    X3+X1       CA[VE] = N$VD */ ASSIGN A *CA*       CCG15     604    A
CGTM              LX6    VD.CAP                                           CCG18      64    A
CGTM              SA6    A3          VDI(C) = VE                          CCG15     605    A
CGTM              SX6    X1+B1       N$VD = N$VD + 1                      CCG15     606    A
CGTM              SA6    A1                                               CCG15     607    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM              EQ     CG$AVO                                                  CCG15        608     A
CGTM      .VD     ENDIF                                                          CCG15        609     A
CGTM      SYM     TITLE  SYM - CONVERT *IH* TO DPC NAME                          CCG15        610     A
CGTM      .T      IFNE   TEST,0                                                  CCG15        611     A
CGTM                                                                             CCG15        612     A
CGTM     **       SYM - RETURN DISPLAY CODED VALUE OF SYMBOL NAME IN X6          CCG15        613     A
CGTM     *                FOR FORTRAN CODED DEBUGGING ROUTINES                  CCG15        614     A
CGTM     *        CALL                                                           CCG15        615     A
CGTM     *                NAM = SYM( IH )                                       CCG15        616     A
CGTM                                                                             CCG15        617     A
CGTM      SYM1    SB1    1                                                       CCG15        618     A
CGTM              RJ     CSN         CONVERT ORDINAL TO BCD NAME                 CCG15        619     A
CGTM              MX5    48                                                      CCG15        620     A
CGTM              BX1    X5*X6                                                   CCG15        621     A
CGTM              RJ     FSN                                                     CCG15        622         I
         -CCG18
CGTM              SA3    =1H                                                     CCG15        623         I
         -CCG18
CGTM              SB2    B3-B1                                                   CCG15        624         I
         -CCG18
CGTM              MX2    B2                                                      CCG15        625         I
         -CCG18
CGTM              BX3    -X2*X3                                                  CCG15        626         I
         -CCG18
CGTM              BX6    X1+X3       SPACE FILL NAME                             CCG15        627         I
         -CCG18
CGTM              CALL   SFN=        SPACE FILL THE NAME                         CCG18         65         I
         -CCG27
CGTM              CALL   ZTB=        REMOVE ZERO BYTES                           CCG27          4     A
CGTM      SYM     ENTRY.                                                         CCG15        628     A
CGTM              SA2    X1+                                                     CCG15        629     A
CGTM              NZ     X2,SYM1                                                 CCG15        630     A
CGTM              SA3    =1H                                                     CCG15        631     A
CGTM              BX6    X3                                                      CCG15        632     A
CGTM              EQ     SYM                                                     CCG15        633     A
CGTM                                                                             CCG15        634     A
CGTM     **       GETNSYM - RETURN N.SYMS IN SYMBOL TABLE                        CCG15        635     A
CGTM                                                                             CCG15        636     A
CGTM      GETNSYM ENTRY.                                                         CCG15        637     A
CGTM              SA2    =XL.SYM                                                 CCG15        638         I
         -CCG18
CGTM              SA2    =XL$SYM                                                 CCG18         66     A
CGTM              SX3    3                                                       CCG15        639     A
CGTM              IX6    X2/X3,B2                                                CCG15        640     A
CGTM              SA6    X1                                                      CCG15        641     A
CGTM              EQ     GETNSYM                                                 CCG15        642     A
CGTM                                                                             CCG15        643     A
CGTM      .T      ENDIF                                                          CCG15        644     A
CGTM      CSN     SPACE  3,14                                                    CCG15        645     A
CGTM     **       CSN - CONVERT SYMTAB *IH* TO BCD NAME                          CCG15        646     A
CGTM     *                                                                       CCG15        647     A
CGTM     *        ENTRY  (X2) = *IH*                                             CCG15        648     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM     *                                                                  CCG15      649    A
CGTM     *         EXIT   (X6) = 0L_BCD-NAME                                CCG15      650    A
CGTM     *                (B3) = BC                                         CCG15      651    A
CGTM     *                                                                  CCG15      652    A
CGTM     *         USES   X - 2, 3, 4, 6   B - 2, 3                         CCG15      653    A
CGTM                                                                        CCG15      654    A
CGTM      CSN0     LX3    B1,X2                                             CCG15      655         I
         -CCG35
CGTM      CSN0     MX3    -13         IH LESS IH.LCMA ETC.          001280  CCG35        7    A
CGTM               BX2    -X3*X2                                    001290  CCG35        8    A
CGTM               LX3    B1,X2                                     001300  CCG35        9    A
CGTM               SB2    X4                                                CCG15      656    A
CGTM               IX3    X2+X3                                             CCG15      657    A
CGTM               SA4    B2+X3       WORDA = SYM(IH)                       CCG15      658    A
CGTM               MX6    -6                                                CCG15      659    A
CGTM               BX3    -X6*X4                                            CCG15      660    A
CGTM               SB3    X3                                                CCG15      661    A
CGTM               BX6    X6*X4                                             CCG15      662    A
CGTM                                                                        CCG15      663    A
CGTM      CSN      ROUTINE                                                   CCG15      664         I
         -CCG16
CGTM      CSN      ENTRY. ,#                                                 CCG16      148    A
CGTM               MX3    -IH.HL                                            CCG15      665    A
CGTM               BX6    X3*X2                                             CCG15      666    A
CGTM               SA4    O.SYM                                             CCG15      667         I
         -CCG16
CGTM               SA4    O$SYM                                             CCG16      149    A
CGTM               ZR     X6,CSN0     IF I[IH] = 0                          CCG15      668    A
CGTM                                                                        CCG15      669    A
CGTM     *         CONVERT FOWARD REFERENCE SYMBOL TO DISPLAY CODE.         CCG15      670    A
CGTM                                                                        CCG15      671    A
CGTM               BX4    -X3*X2                                            CCG15      672    A
CGTM               AX2    IH.HL                                             CCG15      673    A
CGTM               SA3    CSNA-1+X2                                         CCG15      674    A
CGTM               LX4    60-IH.HL                                          CCG15      675    A
CGTM               BX6    X3                                                CCG15      676    A
CGTM               SB3    9*6-1       BC = 54-1                             CCG15      677    A
CGTM                                                                        CCG15      678    A
CGTM     +         LX4    3                                                 CCG15      679    A
CGTM               SB3    B3-6        BC = BC - 6                           CCG15      680    A
CGTM               SX3    X4                                                CCG15      681    A
CGTM               ZR     X3,*-1      SKIP PAST LEADING ZEROS               CCG15      682    A
CGTM                                                                        CCG15      683    A
CGTM               SB2    60-4*6                                            CCG15      684    A
CGTM      CSN1     IX4    X4-X3       REMOVE DIGIT                          CCG15      685    A
CGTM               LX3    B2                                                CCG15      686    A
CGTM               IX6    X6+X3       ADD DIGIT TO STRING                   CCG15      687    A
CGTM               SB2    B2-6                                              CCG15      688    A
CGTM               LX4    3                                                 CCG15      689    A
CGTM               SX3    X4                                                CCG15      690    A
CGTM               NZ     X4,CSN1                                           CCG15      691    A
CGTM                                                                        CCG15      692    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM              MX3    1                                                  CCG15      693    A
CGTM              AX3    B3                                                 CCG15      694    A
CGTM              BX6    X3*X6       REMOVE GARBAGE FROM THE NAME           CCG15      695    A
CGTM              SB3    B3+B1       BC = BC + 1                            CCG15      696    A
CGTM              EQ     CSN                                                CCG15      697    A
CGTM                                                                       CCG15      698    A
CGTM       CSNA   BSS    0                                                  CCG15      699    A
CGTM              ECHO   1,Z=("HC.FRTP")                                    CCG15      700    A
CGTM              DATA   8L;A.00000                                         CCG15      701    A
CGTM       OPR    TITLE  F.RDT - CCG *IL* INSTRUCTION DESCRIPTOR TABLE      CGTM       135    A
CGTM              LIST   -X,-R,-F                                           CGTM       136    A
CGTM              ECHO   2,A=(I,II,III,IV,BSSZ),B=(0,1,2,3,1)               CGTM       137    A
CGTM              NOREF  A                                                  CGTM       138    A
CGTM       A      EQU    B                                                  CGTM       139    A
CGTM                                                                       CGTM       140    A
CGTM       .T     IFNE   TEST,0                                             CGTM       141    A
CGTM              MACRO  OPR,NAME,SS                                        CGTM       142    A
CGTM       TY     SET    SS                                                 CGTM       143    A
CGTM              CON    TY+1+4L_NAME                                       CGTM       144    A
CGTM              ENDM                                                      CGTM       145    A
CGTM                                                                       CGTM       146    A
CGTM              USE    /OPRS/                                             CGTM       147    A
CGTM     *CALL    OPRDEFS                                                   CGTM       148    A
CGTM              USE    *                                                  CGTM       149    A
CGTM       .T     ENDIF                                                     CGTM       150    A
CGTM              LIST   -R,X,F                                             CGTM       151    A
CGTM       OPR    SPACE  4                                                  CGTM       152    A
CGTM     **       OPR - DEFINE *OPR* MACRO TO FORM RLIST DESCRIPTOR TABLE   CGTM       153    A
CGTM                                                                       CGTM       154    A
CGTM     *        DEFINE VALUES FOR USE IN *OPR* MACRO CALLS                CGTM       155    A
CGTM                                                                       CGTM       156    A
CGTM       MC     SET    34-15                                             CGTM       157    A
CGTM              IFEQ   .CPU,76,1                                          CGTM       158    A
CGTM       MC     SET    37-15                                             CGTM       159    A
CGTM              IF     MIC,CPU990,1    IF 990 CPU DEFINED                CCGA092     85    A
CGTM       MC     SET    31-15                                             CCGA092     86    A
CGTM                                                                       CGTM       160    A
CGTM              ECHO   2,X=(COMM,USI,),V=(1,1,0)                          CGTM       161    A
CGTM              NOREF  .X                                                 CGTM       162    A
CGTM       .X     =      V                                                  CGTM       163    A
CGTM                                                                       CGTM       164    A
CGTM     *        SYMBOL DEFINED IF INSTRUCTION DOESN"T DEFINE AN RI        CGTM       165    A
CGTM                                                                       CGTM       166    A
CGTM              ECHO   2,N=(JUMP,UJUMP,STORE,STORE2,REGST,ECJUMP)         CGTM       167    A
CGTM              NOREF  Y.N                                                CGTM       168    A
CGTM       Y.N    EQU    1                                                  CGTM       169    A
CGTM                                                                       CGTM       170    A
CGTM              ECHO   2,N=(LOAD,STORE,JUMP,UJUMP,LOAD2,STORE2,ECJUMP),_______  CGTM  171    A
CGTM     ,V=(4,2,1,1,14B,12B,20B)                                          CGTM       172    A
CGTM              NOREF  Z.N                                                CGTM       173    A
CGTM       Z.N    EQU    V                                                  CGTM       174    A
CGTM                                                                       CCG18       67    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM              ECHO   2,N=(FMA,S,CLR,LDC,LDV,ILD,TLD)                    CCG18        68     A
CGTM              NOREF  ZRP.N                                              CCG18        69     A
CGTM       ZRP.N  = 1                                                       CCG18        70     A
CGTM                                                                        CGTM        175     A
CGTM     *        BITS SET FOR XMT , CON SHIFTS AND PSEUDO INSTRUCTION FIELDS  CGTM      176     A
CGTM                                                                        CGTM        177     A
CGTM              ECHO   2,N=(SXT,XMT,KLS,KRS,EOQ,BOS,EOS,DAR,RS,DEF,LAB,ENT),__  CGTM  178     A
CGTM     ,V=(4,4,2,2,1,1,1,1,1,1,1,1)                                       CGTM        179     A
CGTM              NOREF  S.N                                                CGTM        180     A
CGTM       S.N    EQU    V                                                  CGTM        181     A
CGTM                                                                        CGTM        182     A
CGTM              MACRO  OPR,NAM,SS                                         CGTM        183     A
CGTM              ECHO   1,X=(BD,LSJ,XSP)                                   CGTM        184     A
CGTM       X      SET    0                                                  CGTM        185     A
CGTM       TYPE   SET    SS                                                 CGTM        186     A
CGTM       S      GETARG 23,4,(SS)                                          CGTM        187     A
CGTM       CO     SET    ."S"                                               CGTM        188     A
CGTM     *                              UNSAFE INSTRUCTION                  CGTM        189     A
CGTM       S      GETARG 29,3,(SS)                                          CGTM        190         I
         -CCGA092
CGTM       S      GETARG 28,3,(SS)                                          CCGA092      87     A
CGTM       CO     SET    CO+4*."S"                                          CGTM        191     A
CGTM     *                          FUNCTION UNIT AND TIMES                 CGTM        192     A
CGTM       S      MICRO  MC,2,$SS$                                          CGTM        193     A
CGTM       FT     SET    "S" 0                                              CGTM        194     A
CGTM       JFT    SET    FT/FT                                              CGTM        195     A
CGTM       S      GETARG 40,6,(SS)                                          CGTM        196     A
CGTM       FU1    SET    0                                                  CGTM        197     A
CGTM              IFC    NE,//"S"/,1                                        CGTM        198     A
CGTM       FU1    SET    /FUNIT/"S"                                         CGTM        199     A
CGTM       FU2    SET    FU1                                                CGTM        200     A
CGTM     *                          SECOND FUNCTIONAL UNIT ( 6600 )         CGTM        201     A
CGTM       #MD    IFEQ   .CPU,74                                  ?6600     CGTM        202     A
CGTM       S      GETARG 47,5,(SS)                                          CGTM        203     A
CGTM              IFC    NE,/"S"//,3                                        CGTM        204     A
CGTM       FU2    SET    FU1+1                                              CGTM        205     A
CGTM              IFC    NE,/"S"/FU1/,1                                     CGTM        206     A
CGTM       FU2    SET    /FUNIT/"S"                                         CGTM        207     A
CGTM       #MD    ENDIF                                           '6600     CGTM        208     A
CGTM     *                          INSTRUCTION SIZE                        CGTM        209     A
CGTM       S      GETARG 54,2,(SS)                                          CGTM        210     A
CGTM       PA     SET    "S" 1                                              CGTM        211     A
CGTM       PA     SET    PA/15                                              CGTM        212     A
CGTM       S      GETARG 63,6,(SS)                                          CGTM        213     A
CGTM              IFC    EQ,/"S"/ZEROP/,1                                   CGTM        214         I
         -CCG18
CGTM              IF     DEF,ZRP.NAM,1                                      CCG18        71     A
CGTM       CO     SET    2                                                  CGTM        215     A
CGTM     *                          LOAD / STORE / JUMP BITS                CGTM        216     A
CGTM              IF     DEF,Z."S",1                                        CGTM        217     A
CGTM       LSJ    SET    Z."S"                                              CGTM        218     A
CGTM     *                          ^D - NO RI / RI NOT A DEF               CGTM        219     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM              IFEQ   TYPE,3,2                                                CGTM      220    A
CGTM      BD      SET    1                                                       CGTM      221    A
CGTM              ELSE   2                                                       CGTM      222    A
CGTM              IF     DEF,Y."S",1                                             CGTM      223    A
CGTM      BD      SET    1                                                       CGTM      224    A
CGTM     *                         BOUNDARY MARKER                              CGTM      225    A
CGTM      S       GETARG 70,1,(SS)                                               CGTM      226    A
CGTM              IFC    NE,/"S"//,1                                             CGTM      227    A
CGTM      BD      SET    BD+2                                                    CGTM      228    A
CGTM              IF     DEF,S.NAM,1                                             CGTM      229    A
CGTM      XSP     SET    S.NAM                                                   CGTM      230    A
CGTM     *                                                                       CGTM      231    A
CGTM       VFD 2/TYPE,5/FT,2/1,4/FU1,4/FU2,2/BD,8/LSJ,3/XSP,2/JFT,3/CO,2/PA,23/  CGTM      232    A
CGTM      OPR     ENDM                                                           CGTM      233    A
CGTM      RDT     SPACE  3                                                       CGTM      234    A
CGTM     **       F.RDT - CCG *IL* INSTRUCTION DESCRIPTOR TABLE                  CGTM      235    A
CGTM                                                                             CGTM      236    A
CGTM              ENTRY  F$RDT       REFERENCED AS * F.RDT * WITHIN *CCG*        CGTM      237    A
CGTM      F$RDT   BSS    0                                                       CGTM      238    A
CGTM              LOC    0                                                       CGTM      239    A
CGTM     *CALL    OPRDEFS                                                        CGTM      240    A
CGTM              LOC    *O                                                      CGTM      241    A
CGTM              LIST   F,R                                                     CGTM      242    A
CGTM      .ID     IFNE   HC.ID,3          ^ PL/I                    ?            CGTM      243    A
CGTM     *CALL    CCOMISC                                                        CGTM      244    A
CGTM              ENTRY  ISC=                                                    CGTM      245    A
CGTM      .ID     ENDIF                   ^ PL/I                    '            CGTM      246    A
CGTM      SST#    TITLE  SST# - SHELL SORT TABLE                                 CGTM      247    A
CGTM     ***      SST# - SHELL SORT OF A SINGLE WORD PER ENTRY TABLE.            CGTM      248    A
CGTM     *                                                                       CGTM      249    A
CGTM     *        SST# SORTS A TABLE USING A SHELL SORT TECHNIQUE.               CGTM      250    A
CGTM     *        THE TABLE IS SORTED IN PLACE INTO ASCENDING ORDER ON THE KEY   CGTM      251    A
CGTM     *        KEY(X) = XOR(SHIFT(KEYMASK&X,KEYSHIFT),UPDOWN)                 CGTM      252    A
CGTM     *                                                                       CGTM      253    A
CGTM     *               TIME(SST) @ TIME(RADIX) FOR N @ 250/575, B6 "0/=0       CGTM      254    A
CGTM     *                                                                       CGTM      255    A
CGTM     *        ENTRY  (B7) = ADDRESS OF TABLE TO BE SORTED                    CGTM      256    A
CGTM     *               (B6) = SHIFT COUNT FOR EXTRACTED KEY PRIOR TO COMPARE   CGTM      257    A
CGTM     *               (X0) = MASK TO EXTRACT KEY FROM ENTRY                   CGTM      258    A
CGTM     *               (X1) = NUMBER OF ELEMENTS IN ARRAY.                     CGTM      259    A
CGTM     *               (X5) = ASCENDING/DESENDING SORT MASK ( +0 / -0 )        CGTM      260    A
CGTM     *                                                                       CGTM      261    A
CGTM     *        USES   X - ALL                                                 CGTM      262    A
CGTM     *               B - 2, 3, 4, 5, 6                                       CGTM      263    A
CGTM     *               A - 1, 2, 6, 7.                                         CGTM      264    A
CGTM                                                                             CGTM      265    A
CGTM      SST0    SA2    B2          I = I + 1                                   CGTM      266    A
CGTM              SA7    B5-B4                                                   CGTM      267    A
CGTM              EQ     B2,B3,SST4  IF I > N                                    CGTM      268    A
CGTM                                                                             CGTM      269    A
CGTM      SST1    BX7    X0*X2                                                   CGTM      270    A
CGTM              SA1    B2+B4       J = I - K                                   CGTM      271    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM              LX6    B6,X7                                          CGTM        272    A
CGTM              SB5    B2+B4                                          CGTM        273    A
CGTM              BX3    X6-X5        SI = KEY(T(I))                    CGTM        274    A
CGTM              SB2    B2+1                                           CGTM        275    A
CGTM              LX7    X2                                             CGTM        276    A
CGTM                                                                   CGTM        277    A
CGTM       SST2   BX2    X0*X1                                          CGTM        278    A
CGTM              LX6    B6,X2                                          CGTM        279    A
CGTM              BX2    X6-X5        SJ = KEY(T(J))                    CGTM        280    A
CGTM              IX6    X3-X2                                          CGTM        281    A
CGTM              PL     X6,SST0      IF ELEMENTS IN ORDER              CGTM        282    A
CGTM              BX6    X1                                             CGTM        283    A
CGTM              SA1    B5+B4        J = J - K                         CGTM        284    A
CGTM              SA6    B5-B4        T(J+K) = T(J)                     CGTM        285    A
CGTM              SB5    B5+B4                                          CGTM        286    A
CGTM              GE     B5,B7,SST2   IF J \ 0                          CGTM        287    A
CGTM              SA2    B2           I = I + 1                         CGTM        288    A
CGTM              SA7    B5-B4        T(J+K) = S                        CGTM        289    A
CGTM              LT     B2,B3,SST1   IF I @ LENGTH                     CGTM        290    A
CGTM                                                                   CGTM        291    A
CGTM       SST4   AX4    1            K = K/2                           CGTM        292    A
CGTM              SB4    X4                                             CGTM        293    A
CGTM              SA2    B7-B4        I = K                             CGTM        294    A
CGTM              SB2    B7-B4                                          CGTM        295    A
CGTM              NZ     X4,SST1      IF K " 0                          CGTM        296    A
CGTM       SST    ENTRY. **,#                                          CGTM        297    A
CGTM              MX4    12                                             CGTM        298    A
CGTM              NX6    B2,X1        K = 2**FIX(LOG2(LENGTH))+1        CGTM        299    A
CGTM              SB1    1                                              CGTM        300    A
CGTM              SB3    B7+X1        (B3) = LWA + 1                    CGTM        301    A
CGTM              AX4    B2,X4                                          CGTM        302    A
CGTM              NZ     B6,SST4      IF SHFTC " 0                      CGTM        303    A
CGTM                                                                   CGTM        304    A
CGTM              QUAL   SHFTC=0                                        CGTM        305    A
CGTM                                                                   CGTM        306    A
CGTM       SST0   AX4    1                                              CGTM        307    A
CGTM              SB4    X4           K = K/2                           CGTM        308    A
CGTM              SA2    B7-B4                                          CGTM        309    A
CGTM              SB2    B7-B4        I = K                             CGTM        310    A
CGTM              NZ     X4,SST2      IF K " 0                          CGTM        311    A
CGTM              EQ     SST                                            CGTM        312    A
CGTM                                                                   CGTM        313    A
CGTM       SST1   SA2    A2+B1        I = I + 1                         CGTM        314    A
CGTM              SA7    B5-B4        T(J+K) = S                        CGTM        315    A
CGTM              EQ     B2,B3,SST0   IF I > N                          CGTM        316    A
CGTM                                                                   CGTM        317    A
CGTM       SST2   BX6    X0*X2                                          CGTM        318    A
CGTM              SA1    B2+B4        J = I - K                         CGTM        319    A
CGTM              LX7    X2           S = T(I)                          CGTM        320    A
CGTM              SB5    B2+B4                                          CGTM        321    A
CGTM              BX3    X6-X5        SI = KEY(T(I))                    CGTM        322    A
CGTM              SB2    B2+1                                           CGTM        323    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM              BX2    X0*X1                                          CGTM        324    A
CGTM                                                                    CGTM        325    A
CGTM      SST3    BX2    X2-X5                                          CGTM        326    A
CGTM              IX6    X3-X2                                          CGTM        327    A
CGTM              PL     X6,SST1     IF ELEMENTS IN ORDER               CGTM        328    A
CGTM              LX6    X1                                             CGTM        329    A
CGTM              SA1    B5+B4                                          CGTM        330    A
CGTM              SA6    B5-B4       T(J+K) = T(J)                      CGTM        331    A
CGTM              SB5    B5+B4       J = J - K                          CGTM        332    A
CGTM              BX2    X0*X1                                          CGTM        333    A
CGTM              GE     B5,B7,SST3                                     CGTM        334    A
CGTM              SA2    A2+B1       I = I + 1                          CGTM        335    A
CGTM              SA7    B5-B4                                          CGTM        336    A
CGTM              LT     B2,B3,SST2  IF I @ N                           CGTM        337    A
CGTM              EQ     SST0                                           CGTM        338    A
CGTM                                                                    CGTM        339    A
CGTM              QUAL                                                   CGTM        340    A
CGTM      SHL#    TITLE  SHL# - SHELL SORT                              CGTM        341    A
CGTM     **       SHL# - SHELL SORT FOR A 1 WORD PER ENTRY TABLES.      CGTM        342    A
CGTM     *        INPLACE SORT OF TABLE INTO ASCENDING ORDER            CGTM        343    A
CGTM     *                                                              CGTM        344    A
CGTM     *               TIME(SHL) @ TIME(RADIX) FOR N @ 1650           CGTM        345    A
CGTM     *                                                              CGTM        346    A
CGTM     *        ENTRY  (X1) = N = TABLE LENGTH                        CGTM        347    A
CGTM     *               (B7) = TBL = = FWA OF ARRAY TO BE SORTED       CGTM        348    A
CGTM     *                                                              CGTM        349    A
CGTM     *        EXIT   (B3) = LWA+1 OF TABLE                          CGTM        350    A
CGTM     *               (B7) = FWA                                     CGTM        351    A
CGTM     *                                                              CGTM        352    A
CGTM     *        USES   X - 1, 2, 3, 4, 6, 7                           CGTM        353    A
CGTM     *               B - 2, 3, 4, 5                                 CGTM        354    A
CGTM     *               A - 1, 2, 6, 7                                 CGTM        355    A
CGTM                                                                    CGTM        356    A
CGTM      SHL1    SA2    A2+B1       I = I + 1                          CGTM        357    A
CGTM              SA7    B5-B4       T(J+K) = S                         CGTM        358    A
CGTM              EQ     B2,B3,SHL4  IF I > N                           CGTM        359    A
CGTM                                                                    CGTM        360    A
CGTM      SHL2    SA1    B2+B4       J = I - K                          CGTM        361    A
CGTM              LX7    X2          S = T(I)                           CGTM        362    A
CGTM              SB5    B2+B4                                          CGTM        363    A
CGTM              SB2    B2+B1                                          CGTM        364    A
CGTM                                                                    CGTM        365    A
CGTM      SHL3    IX3    X2-X1                                          CGTM        366    A
CGTM              BX6    X1                                             CGTM        367    A
CGTM              PL     X3,SHL1     IF ELEMENTS IN ORDER               CGTM        368    A
CGTM              SA1    A1+B4                                          CGTM        369    A
CGTM              NO                                                     CGTM        370    A
CGTM              SA6    B5-B4       T(J+K) = T(J)                      CGTM        371    A
CGTM              SB5    B5+B4       J = J - K                          CGTM        372    A
CGTM              GE     B5,B7,SHL3  IF J \ 0                           CGTM        373    A
CGTM              EQ     SHL1                                           CGTM        374    A
CGTM                                                                    CGTM        375    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM      SHL4     AX4    1                                                  CGTM        376     A
CGTM               SB4    X4          K = K/2                                CGTM        377     A
CGTM               SA2    B7-B4       I = K                                  CGTM        378     A
CGTM               SB2    B7-B4                                              CGTM        379     A
CGTM               NZ     X4,SHL2     IF K " 0                               CGTM        380     A
CGTM                                                                         CGTM        381     A
CGTM      SHL      ENTRY. **,#                                               CGTM        382     A
CGTM               MX4    12                                                 CGTM        383     A
CGTM               SB1    1                                                  CGTM        384     A
CGTM               NX6    B2,X1       K = 2**(FIX(LOG2(N))+1)                CGTM        385     A
CGTM               SB3    B7+X1       (B3) = LWA+1                           CGTM        386     A
CGTM               AX4    B2,X4                                              CGTM        387     A
CGTM               EQ     SHL4                                               CGTM        388     A
CGTM      TABLES   TITLE  CCG TABLES                                         CGTM        389     A
CGTM               USE    //                                                 CGTM        390         I
          -CCG16
CGTM      F.MEM    BSS    1           FWA OF WORKING STORAGE                 CGTM        391         I
          -CCG16
CGTM               USE    0                                                  CGTM        392         I
          -CCG16
CGTM                                                                         CGTM        393         I
          -CCG16
CGTM      **       TABLE - GENERATE MANAGED TABLE.                            CGTM        394         I
          -CCG16
CGTM      *                                                                  CGTM        395         I
          -CCG16
CGTM      *        TABLE  TNAME,EQUIV                                         CGTM        396         I
          -CCG16
CGTM      *        ENTRY  *TNAME* = TABLE NAME.                               CGTM        397         I
          -CCG16
CGTM      *                *EQUIV* = EQUIVALENT TABLE NAME.  THIS ALLOWS CERTAIN  CGTM  398         I
          -CCG16
CGTM      *                          TABLES TO BE USED BY DIFFERENT PROCESSORS.  CGTM  399         I
          -CCG16
CGTM      *        EXIT   *O$TNAM* = NAME OF WORD CONTAINING THE TABLE FWA    CGTM        400         I
          -CCG16
CGTM      *                *L$TNAM* = NAME OF WORD CONTAINING THE TABLE LENGTH.  CGTM     401         I
          -CCG16
CGTM      *                *Z$TNAM* = TABLE NUMBER                            CGTM        402         I
          -CCG16
CGTM      *        NOTE THAT THE CODE IN *CCG* REFERENCES THE POINTERS BY THE CGTM        403         I
          -CCG16
CGTM      *        NAMES *O.TNAM* AND *L.TNAM* .                              CGTM        404         I
          -CCG16
CGTM                                                                         CGTM        405         I
          -CCG16
CGTM               PURGMAC TABLE                                              CGTM        406         I
          -CCG16
CGTM               MACRO  TABLE,TNAM,EQUIV                                    CGTM        407         I
          -CCG16
CGTM               ENTRY  O$TNAM,L$TNAM,Z$TNAM                                CGTM        408         I
          -CCG16

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM     T         IFC    EQ,/EQUIV//                                          CGTM        409         I
         -CCG16
CGTM      Z$TNAM   EQU    *-FTAB                                               CGTM        410         I
         -CCG16
CGTM      O$TNAM   CON    F.MEM                                                CGTM        411         I
         -CCG16
CGTM      TABLES   RMT                                                         CGTM        412         I
         -CCG16
CGTM               ORG    LTAB+Z$TNAM                                          CGTM        413         I
         -CCG16
CGTM      L$TNAM   CON    0                                                    CGTM        414         I
         -CCG16
CGTM      TABLES   RMT                                                         CGTM        415         I
         -CCG16
CGTM     T         ELSE                                                         CGTM        416         I
         -CCG16
CGTM      Z$TNAM   EQU    Z$EQUIV                                              CGTM        417         I
         -CCG16
CGTM      O$TNAM   EQU    O$EQUIV                                              CGTM        418         I
         -CCG16
CGTM      TABLES   RMT                                                         CGTM        419         I
         -CCG16
CGTM      L$TNAM   EQU    L$EQUIV                                              CGTM        420         I
         -CCG16
CGTM      TABLES   RMT                                                         CGTM        421         I
         -CCG16
CGTM      T        ENDIF                                                       CGTM        422         I
         -CCG16
CGTM      TABNAM   RMT                                                         CGTM        423         I
         -CCG16
CGTM               ORG    NAMTAB+Z$TNAM                                        CGTM        424         I
         -CCG16
CGTM               DATA   10H TNAM EQUIV                                       CGTM        425         I
         -CCG16
CGTM      TABNAM   RMT                                                         CGTM        426         I
         -CCG16
CGTM               ENDM                                                        CGTM        427         I
         -CCG16
CGTM      TABLES   SPACE  3                                                    CGTM        428         I
         -CCG16
CGTM     **        PASS 2 MANAGED TABLE DEFINITIONS                            CGTM        429         I
         -CCG16
CGTM                                                                           CGTM        430         I
         -CCG16
CGTM      FTAB     BSS    0           FWA VECTOR FOR TABLE MANAGER             CGTM        431         I
         -CCG16
CGTM                                                                           CGTM        432         I
         -CCG16
CGTM      BLK      TABLE              BLOCKS ( OPT=2 ONLY )                    CGTM        433         I
         -CCG16
CGTM                                                                           CGTM        434         I
         -CCG16

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM      TXT      TABLE              RLIST INSTRUCTIONS                          CGTM      435      I
          -CCG16
CGTM      GST      TABLE  TXT         GRAPH STRUCTURE TABLE                       CGTM      436      I
          -CCG16
CGTM                                                                              CGTM      437      I
          -CCG16
CGTM      RND      TABLE              R-NUMBER DEFINITION TABLE ( *SQZ* )         CGTM      438      I
          -CCG16
CGTM      TREE     TABLE  RND         DEPENDENCY TREE / SUCCESSOR INDEX TABLE     CGTM      439      I
          -CCG16
CGTM                                                                              CGTM      440      I
          -CCG16
CGTM      PIT      TABLE              POSTED INSTRUCTIONS                         CGTM      441      I
          -CCG16
CGTM                                                                              CGTM      442      I
          -CCG16
CGTM      OTI      TABLE              OT. INFO 12/P(OT. ORD),30/0,18/RI IN OT.    CGTM      443      I
          -CCG16
CGTM      MLT      TABLE  OTI         MOD LIST INDEX TABLE                        CGTM      444      I
          -CCG16
CGTM                                                                              CGTM      445      I
          -CCG16
CGTM      MOD      TABLE              MODIFICATIONS ( INSTRUCTIONS )              CGTM      446      I
          -CCG16
CGTM                                                                              CGTM      447      I
          -CCG16
CGTM      CFT      TABLE              CONTROL FLOW TABLE ( EDGES )                CGTM      448      I
          -CCG16
CGTM                                                                              CGTM      449      I
          -CCG16
CGTM      BST      TABLE              BLOCK STATUS INFORMATION                    CGTM      450      I
          -CCG16
CGTM      * BIT    =      BST         ( PHASE 1 OF OPT=2 )                        CGTM      451      I
          -CCG16
CGTM                                                                              CGTM      452      I
          -CCG16
CGTM      PSI      TABLE              POST STORE INFORMATION ( *GPO/GRA* )        CGTM      453      I
          -CCG16
CGTM                                                                              CGTM      454      I
          -CCG16
CGTM      RXI      TABLE              REGION EXIT INFORMATION ( *GPO/GRA* )       CGTM      455      I
          -CCG16
CGTM                                                                              CGTM      456      I
          -CCG16
CGTM      RCT      TABLE              REGISTER CANIDATE TABLE ( *GRA* )           CGTM      457      I
          -CCG16
CGTM                                                                              CGTM      458      I
          -CCG16
CGTM      TET      TABLE              TEMP EQUIVALENCE TABLE ( CODE MOTION )      CGTM      459      I
          -CCG16
CGTM                                                                              CGTM      460      I
          -CCG16

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM      IOL      TABLE                I/O AP LIST PARAMETER ORDINALS ( OPT=2 )          CGTM      461      I
          -CCG16
CGTM      IIT      TABLE  IOL           INCREMENT INFORMATION ( OPT=2 )                   CGTM      462      I
          -CCG16
CGTM                                                                                      CGTM      463      I
          -CCG16
CGTM      UDI      TABLE                USE/DEF INDEX TABLE                               CGTM      464      I
          -CCG16
CGTM                                                                                      CGTM      465      I
          -CCG16
CGTM               IFC    NE,//"HC.HDT"/,2                                                CGTM      466      I
          -CCG16
CGTM               ECHO   1,Z=("HC.HDT")                                                  CGTM      467      I
          -CCG16
CGTM      Z        TABLE                                                                  CGTM      468      I
          -CCG16
CGTM                                                                                      CGTM      469      I
          -CCG16
CGTM      CVT      TABLE                CONSTANT VALUE TABLE                              CGTM      470      I
          -CCG16
CGTM                                                                                      CGTM      471      I
          -CCG16
CGTM      CUT      TABLE                CONSTANT USED TABLE                               CGTM      472      I
          -CCG16
CGTM                                                                                      CGTM      473      I
          -CCG16
CGTM      N.DT     EQU    *-FTAB        N. DYNAMIC TABLES                                 CGTM      474      I
          -CCG16
CGTM                                                                                      CGTM      475      I
          -CCG16
CGTM               ECHO   1,Z=("HC.FRTP")                                                  CGTM      476      I
          -CCG16
CGTM      ;AT      TABLE                 ;A ADDRESS DEFINITION TABLE                      CGTM      477      I
          -CCG16
CGTM      SST      TABLE                SST - SET SYMBOL VALUE TABLE                      CGTM      478      I
          -CCG16
CGTM                                                                                      CGTM      479      I
          -CCG16
CGTM      CBT      TABLE                COMMON BLOCK NAME AND LENGTH TABLE                CGTM      480      I
          -CCG16
CGTM      *                             60/0L_NAME,2/LVL,34/,24/LENGTH                    CGTM      481      I
          -CCG16
CGTM                                                                                      CGTM      482      I
          -CCG16
CGTM      LBT      TABLE                LOCAL BLOCK NAME AND LENGTH TABLE                 CGTM      483      I
          -CCG16
CGTM      *                             42/7L_NAME,18/LENGTH                              CGTM      484      I
          -CCG16
CGTM                                                                                      CGTM      485      I
          -CCG16
CGTM               IFNE   HC.QBT,0,1                                                      CGTM      486      I
          -CCG16

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM      QBT      TABLE              QUAL BLOCK NAME TABLE                          CGTM        487        I
          -CCG16
CGTM                                                                                CGTM        488        I
          -CCG16
CGTM               IFC    NE,/"HC.MCT"//,2                                          CGTM        489        I
          -CCG16
CGTM               ECHO   1,Z=("HC.MCT")                                            CGTM        490        I
          -CCG16
CGTM      Z        TABLE              Z TABLE                                        CGTM        491        I
          -CCG16
CGTM                                                                                CGTM        492        I
          -CCG16
CGTM      SYM      TABLE              SYMBOL TABLE                                   CGTM        493        I
          -CCG16
CGTM               BSS    0                                                         CGTM        494        I
          -CCG16
CGTM      NTAB     EQU    *-FTAB      TOTAL NUMBER OF TABLES                         CGTM        495        I
          -CCG16
CGTM               SPACE  3                                                         CGTM        496        I
          -CCG16
CGTM      **       REMAINDER OF TABLE MANAGER STORAGE                                CGTM        497        I
          -CCG16
CGTM                                                                                CGTM        498        I
          -CCG16
CGTM      LTAB     BSS    0           TABLE LENGTHS                                  CGTM        499        I
          -CCG16
CGTM      TABLES   HERE                                                             CGTM        500        I
          -CCG16
CGTM      **       TABLE DEFINE MANAGED TABLE FOR CCG                                CCG16       150      A
CGTM                                                                                CCG16       151      A
CGTM               MACRO  TABLE,TNAM,EQUIV                                          CCG16       152      A
CGTM      TOC      IFEQ   HC.2OC,0                                                  CCG16       153      A
CGTM      TNAM     B$TBL  EQUIV                                                     CCG16       154      A
CGTM      .T       SET    Z$TNAM+1                                                  CCG18        72      A
CGTM      TOC      ELSE                                                             CCG16       155      A
CGTM               EXT    O$TNAM,L$TNAM                                             CCG16       156      A
CGTM      FF       IFEQ   .T,0                                                      CCG16       157      A
CGTM      FTAB     EQU    O$TNAM                                                    CCG16       158      A
CGTM      LTAB     EQU    L$TNAM                                                    CCG16       159      A
CGTM      FF       ENDIF                                                            CCG16       160      A
CGTM      EE       IFC    EQ,/EQUIV//                                               CCG16       161      A
CGTM      Z$TNAM   EQU    .T                                                        CCG16       162      A
CGTM      .T       SET    .T+1                                                      CCG16       163      A
CGTM      EE       ELSE                                                             CCG16       164      A
CGTM      Z$TNAM   EQU    Z$EQUIV                                                   CCG16       165      A
CGTM      EE       ENDIF                                                            CCG16       166      A
CGTM      TOC      ENDIF                                                            CCG16       167      A
CGTM      TABNAM   RMT                                                              CCG16       168      A
CGTM               ORG    NAMTAB+Z$TNAM                                             CCG16       169      A
CGTM               DATA   10H TNAM EQUIV                                            CCG16       170      A
CGTM      TABNAM   RMT                                                              CCG16       171      A
CGTM               ENDM                                                             CCG16       172      A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM                                                                                    CCG16      173    A
CGTM      .T       SET    0                                                             CCG16      174    A
CGTM                                                                                    CCG16      175    A
CGTM               IFEQ   HC.2OC,0,1                                                    CCG16      176    A
CGTM      FTAB     BSS    0                 FWA TABLE VECTOR                            CCG16      177    A
CGTM                                                                                    CCG16      178    A
CGTM     *CALL     CCGTMTV                                                              CCG16      179    A
CGTM                                                                                    CCG16      180    A
CGTM      NTAB     EQU    .T          N. TOTAL TABLES                                  CCG16      181    A
CGTM                                                                                    CCG16      182    A
CGTM               IFEQ   HC.2OC,0                                                      CCG16      183    A
CGTM      LTAB     BSS    0                                                            CCG16      184    A
CGTM      TABLES   HERE                                                                CCG16      185    A
CGTM               USE    //                                                           CCG16      186         I
         -CCG18
CGTM      F.MEM    BSS    1                                                            CCG16      187         I
         -CCG18
CGTM               EXT    F.MEM                                                        CCG18       73    A
CGTM               USE    *                                                            CCG16      188    A
CGTM                                                                                    CCG16      189    A
CGTM               ELSE                                                                 CCG16      190    A
CGTM      F.MEM    EQU    0                                                            CCG16      191    A
CGTM               ENDIF                                                                CCG16      192    A
CGTM                                                                                    CCG16      193    A
CGTM               ENTRY  F$FRT                                                        CCG16      194    A
CGTM      F$FRT    BSS    0           FOWARD REFERENCE POINTER TABLE                   CCG16      195    A
CGTM               ECHO   1,Z=("HC.FRTP")                                              CCG16      196    A
CGTM               VFD    42/,18/=XO$Z_T                                               CCG16      197    A
CGTM               IFNE   HC.IA,0,1                                                    CCG18       74    A
CGTM               VFD    42/,18/=XF$SST                                               CCG16      198    A
CGTM                                                                                    CGTM       501    A
CGTM      LM       ENTRY. F.MEM,#     LOW MEMORY ADDRESS ( FWA WORKING STORAGE )       CGTM       502    A
CGTM      TN       CON    N.DT+1+HC.NDAT    NUMBER OF ACTIVE TABLES                    CGTM       503         I
         -CCG16
CGTM      TN       CON    N.DT+1      NUMBER OF ACTIVE TABLES                          CCG16      199    A
CGTM               BSS    4           SCRATCH TO SAVE - X0 , X5 , X1 , A0              CGTM       504    A
CGTM      PL       CON    0           PREVIOUS LENGTH OF REQUESTING TABLE              CGTM       505    A
CGTM      TO       ENTRY. TOV#,#      ADDRESS OF TABLE OVERFLOW PROCESSOR              CGTM       506    A
CGTM      MX.BLN   ENTRY. 0,#                                                          CGTM       507    A
CGTM      THRESH   ENTRY. 6000B,#                                                      CCGA011     13    A
CGTM               SPACE  3                                                             CGTM       508         I
         -CCG15
CGTM     *         TABLE DECLARATIONS FOR THE ASSEMBLER                                CGTM       509         I
         -CCG15
CGTM                                                                                    CGTM       510         I
         -CCG15
CGTM               PURGMAC TABLE                                                        CGTM       511         I
         -CCG15
CGTM               MACRO  TABLE,N                                                       CGTM       512         I
         -CCG15
CGTM               ENTRY  O$N,L$N,Z$N                                                   CGTM       513         I
         -CCG15

         0        1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM      Z$N      =      .T                                                    CGTM      514      I
          -CCG15
CGTM      O$N      =      FTAB+.T                                               CGTM      515      I
          -CCG15
CGTM      L$N      =      LTAB+.T                                               CGTM      516      I
          -CCG15
CGTM      .T       SET    .T+1                                                  CGTM      517      I
          -CCG15
CGTM               ENDM                                                         CGTM      518      I
          -CCG15
CGTM                                                                            CGTM      519      I
          -CCG15
CGTM               NOREF  .T                                                    CGTM      520      I
          -CCG15
CGTM      .T       SET    1                                                     CGTM      521      I
          -CCG15
CGTM                                                                            CGTM      522      I
          -CCG15
CGTM               IFNE   HC.FPAS,0,1                                           CGTM      523      I
          -CCG15
CGTM      FPST     TABLE              F.P. SUB WORDS                            CGTM      524      I
          -CCG15
CGTM                                                                            CGTM      525      I
          -CCG15
CGTM      LAFT     TABLE              LINK AND FILL TABLE                       CGTM      526      I
          -CCG15
CGTM                                                                            CGTM      527      I
          -CCG15
CGTM      XFT      TABLE              EXTENDED FILL TABLE                       CGTM      528      I
          -CCG15
CGTM                                                                            CGTM      529      I
          -CCG15
CGTM      XLT      TABLE              EXTENDED LINK TABLE                       CGTM      530      I
          -CCG15
CGTM                                                                            CGTM      531      I
          -CCG15
CGTM      LNT      TABLE              LINE NUMBER TABLE                         CGTM      532      I
          -CCG15
CGTM                                                                            CGTM      533      I
          -CCG15
CGTM      PIDL     TABLE              PIDL ( 34 ) TABLE                         CGTM      534      I
          -CCG15
CGTM                                                                            CGTM      535      I
          -CCG15
CGTM      LBIT     TABLE              LOCAL BLOCK INFO TABLE                    CGTM      536      I
          -CCG15
CGTM                                                                            CGTM      537      I
          -CCG15
CGTM      ENTR     TABLE              ENTRY POINTS ( 36 ) TABLE                 CGTM      538      I
          -CCG15
CGTM                                                                            CGTM      539      I
          -CCG15

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM      LCC      TABLE              LCC DIRECTIVES TABLE                          CGTM         540          I
          -CCG15
CGTM                                                                               CGTM         541          I
          -CCG15
CGTM      LIB      TABLE              LIBRARY NAMES                                 CGTM         542          I
          -CCG15
CGTM                                                                               CGTM         543          I
          -CCG15
CGTM               PURGMAC TABLE                                                   CGTM         544          I
          -CCG15
CGTM      TOV      TITLE  TABLE OVERFLOW PROCESSING ROUTINES                       CGTM         545     A
CGTM     **        TOV - TABLE OVERFLOW PROCESSOR                                  CGTM         546     A
CGTM                                                                               CGTM         547     A
CGTM               ENTRY  TOV#,TOVA#                                               CGTM         548          I
          -CCG18
CGTM      .IA      IFNE   HC.IA,0                                                  CCG18         75     A
CGTM               ENTRY  TOVA#                                                    CCG18         76     A
CGTM      TOVA#    SX0    B5                                                       CGTM         549     A
CGTM               CALL   DAT#        RETURNS (B5) = SPACE FREED UP                CGTM         550     A
CGTM               SB5    B5+X0                                                   CGTM         551     A
CGTM               PL     B5,ATS9     IF ENOUGH SPACE AVAILABLE                   CGTM         552          I
          -CCGA011
CGTM               PL     B5,ATS65    IF ENOUGH SPACE AVAILABLE                   CCGA011       14     A
CGTM                                                                               CGTM         553          I
          -CCG18
CGTM      TOV#     BSS    0                                                       CGTM         554          I
          -CCG18
CGTM      TOV      SB5    -B5                                                     CGTM         555          I
          -CCG18
CGTM      .IA      ENDIF                                                          CCG18         77     A
CGTM                                                                               CCG18         78     A
CGTM               ENTRY  TOV#                                                    CCG18         79     A
CGTM      TOV#     SB5    -B5                                                     CCG18         80     A
CGTM               SX1    B5+2000B    WR = NO.OF WORDS REQUESTED                  CGTM         556     A
CGTM               PRINT  TOV,(* TOV CALLS MOREFL,WR=*Z7),(X1)                    CGTM         557     A
CGTM               CALL   MOREFL                                                  CGTM         558     A
CGTM               PL     X7,ATS9                                                 CGTM         559          I
          -CCGA011
CGTM               PL     X7,ATS65                                                CCGA011       15     A
CGTM      .T       IFNE   TEST,0                                  * TEST MODE *   CGTM         560     A
CGTM               SA5    ATS                                                     CGTM         561     A
CGTM               LX5    30                                                      CGTM         562     A
CGTM      TOV      SNAP   FTAB,TO                                                 CGTM         563     A
CGTM      .T       ENDIF                                          * TEST MODE *   CGTM         564     A
CGTM                                                                               CGTM         565     A
CGTM      TOV1     SA1    =7LTBL-OVF                                              CGTM         566     A
CGTM               CALL   PUNT        ISSUE AN ERROR MESSAGE AND QUIT              CGTM         567     A
CGTM      TOB      SPACE  3,14                                                    CGTM         568     A
CGTM     **        TOB - OVERFLOW PROCESSING FOR PASS2,OPT2                       CGTM         569     A
CGTM     *                                                                         CGTM         570     A
CGTM     *         ENTRY (B5) = - NO.OF WORDS NEEDED                              CGTM         571     A
CGTM     *                                                                         CGTM         572     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM     *         TOB ATTEMPTS TO OBTAIN NEEDED WORDS BY DECREASING BLK TABLE          CGTM        573     A
CGTM     *         IF IT IS AT LEAST 1.5 TIMES THE MAX.BASIC BLOCK SIZE,ELSE            CGTM        574     A
CGTM     *         TOB CALLS MOREFL TO GET ADDIT.SPACE BY EXTENDING FIELD               CGTM        575     A
CGTM     *         LENGTH. IF NO FL AVAIL.,COMPILATION IS ABORTED                       CGTM        576     A
CGTM                                                                                    CGTM        577     A
CGTM               ENTRY  TOB#                                                          CGTM        578     A
CGTM      TOB#     SA2    MX.BLN                                                        CGTM        579     A
CGTM               PRINT  TOB,(* OVERFLOW, WORDS NEEDED = *Z7),(B5)                     CCGA011      16     A
CGTM               SA3    =XM.BBL#                                                      CGTM        580     A
CGTM               AX2    1           X2 = MX.BLN/2                                     CGTM        581     A
CGTM               BX1    X3                                                            CGTM        582     A
CGTM               AX1    1                                                             CGTM        583     A
CGTM               IX1    X1+X3       X1 = 1.5*M.BBL                                    CGTM        584     A
CGTM               IX4    X2-X1                                                         CGTM        585     A
CGTM               PL     X4,TOB1     IF MX.BLN/2 > 1.5*M.BBL                           CGTM        586     A
CGTM               BX2    X1                                                            CGTM        587     A
CGTM                                                                                    CGTM        588     A
CGTM     *         MIN.BLK = MAX(MX.BLN/2,1.5*M.BBL)                                    CGTM        589     A
CGTM                                                                                    CGTM        590     A
CGTM      TOB1     SB5    -B5         WN = NO.OF WORDS NEEDED                           CGTM        591     A
CGTM               PRINT  TOB,(*  NO.OF WORDS NEEDED,MIN.BLK = *2Z7),(B5,X1)            CGTM        592     A
CGTM               SA3    L$BLK                                                         CGTM        593     A
CGTM               SX1    B5+200B     WR = WN+200B */ WORDS TO BE REQUESTED             CGTM        594     A
CGTM               IX4    X3-X2       WA = L.BLK - MIN.BLK */ WORDS AVAIL.IN BLK        CGTM        595     A
CGTM               SB2    B1          SPECIAL PROC.BY DMB IF MOREFL NOT CALLED          CGTM        596     A
CGTM               IX2    X4-X1                                                         CGTM        597     A
CGTM               BX6    X1                                                            CGTM        598     A
CGTM               PL     X2,TOB2     IF WA > WR                                        CGTM        599         I
          -CCGA027
CGTM               SB3    A0
>>>>      CCGA027 12                                                                    CCGA027       6     A
CGTM               ZR     B3,TOB1A    IF ALLOC BLK
>>>>      CCGA027 13                                                                    CCGA027       7     A
CGTM               PL     X2,TOB2     IF WA GT WR
>>>>      CCGA027 14                                                                    CCGA027       8     A
CGTM
>>>>      CCGA027 15                                                                    CCGA027       9     A
CGTM       TOB1A   BSS    0
>>>>      CCGA027 16                                                                    CCGA027      10     A
CGTM               CALL   MOREFL      MOREFL(X6) = WG */ WORDS GRANTED IN NEW FL        CGTM        600     A
CGTM               PL     X7,ATS9     IF WG > WR                                        CGTM        601         I
          -CCGA011
CGTM               PL     X7,ATS65    IF WG > WR                                        CCGA011      17     A
CGTM                                                                                    CGTM        602     A
CGTM               IX6    X1-X6       WR = WR-WG                                        CGTM        603     A
CGTM               SB2    B0          STAND.PROC BY DMB IF MOREFL CALLED                CGTM        604     A
CGTM                                                                                    CGTM        605     A
CGTM     *         GET REMAINING NEEDED SPACE BY REDUCING *BLK* SIZE                    CGTM        606     A
CGTM                                                                                    CGTM        607     A
CGTM       TOB2    SA2    =XMX.AVS#                                                     CGTM        608     A
CGTM               IX1    X6-X2       WD = WR - MX.AVS */ WORDS TO BE DUMPED            CGTM        609     A
CGTM               SB2    X1                                                            CGTM        610     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM              LE     B2,B0,TOB3  IF WD .LE. ZERO                                  CGTM        611     A
CGTM              SA6    TOBA        SAVE WR                                          CGTM        612     A
CGTM              PRINT  TOB,(*  TOB CALLS  DMB ; WDS,SP.FLG=*2Z7),(X1,B2)            CGTM        613     A
CGTM              CALL   DMB#        DUMP BLOCKS TO FREE *WD* WORDS                   CGTM        614     A
CGTM     *        DMB(X2) = WFL */ NO. OF WORDS BY MOREFL                             CGTM        615     A
CGTM              NZ     X1,TOV1     IF X1 " 0 */ CANNOT DUMP ENOUGH BLOCKS           CGTM        616     A
CGTM              SA1    TOBA        RELOAD WR                                        CGTM        617     A
CGTM              IX6    X1-X2       BLK.DECR = WR - WFL                              CGTM        618     A
CGTM              MI     X6,TOB4     IF BLK.DECR @ 0                                  CGTM        619     A
CGTM                                                                                  CGTM        620     A
CGTM     *        DECREASE BLOCK TABLE SIZE BY BLK.DECR                               CGTM        621     A
CGTM                                                                                  CGTM        622     A
CGTM      TOB3    SA1    L$BLK                                                        CGTM        623     A
CGTM              IX7    X1-X6                                                        CGTM        624     A
CGTM              SA7    A1          L.BLK = L.BLK - WR                               CGTM        625     A
CGTM              PRINT  TOB,(*  NEW BLK SIZE,DIFFERENCE =*2Z7),(X7,X6)               CGTM        626     A
CGTM      TOB4    CALL   CGB#        COLLECT FREE SPACE AT THE END OF *BLK*           CGTM        627     A
CGTM              EQ     ATS9                                                         CGTM        628           I
         -CCGA011
CGTM              EQ     ATS65                                                        CCGA011      18     A
CGTM                                                                                  CGTM        629     A
CGTM      TOBA    BSS    1                                                            CGTM        630     A
CGTM      MOREFL  SPACE  3,14                                                         CGTM        631     A
CGTM     **       GET MORE MEMORY FROM THE SYSTEM                                     CGTM        632     A
CGTM     *                                                                            CGTM        633     A
CGTM     *        ENTRY (X1) = WN ,  NO.OF WORDS NEEDED                               CGTM        634     A
CGTM     *                                                                            CGTM        635     A
CGTM     *        EXIT   (X1) = WN                                                    CGTM        636     A
CGTM     *               (X6) = WG , NO. OF WORDS GRANTED                             CGTM        637     A
CGTM     *               (X7) = WG - WN                                               CGTM        638     A
CGTM                                                                                  CGTM        639     A
CGTM      MOREFL  ENTRY. **,#                                                         CGTM        640     A
CGTM              PRINT  MOREFL,(*  WN =*Z7),(X1)                                     CGTM        641     A
CGTM              SA2    HO$MFLS     MXFL = HO$MFLS  */ MAX FL THAT WE CAN GET        CGTM        642     A
CGTM              SA3    =XCP.AFLS   CFL = FL CURRENTLY USED                          CGTM        643     A
CGTM              AX2    30                                                           CGTM        644     A
CGTM              IX6    X2-X3       WA = NO. OF WORDS AVAILABLE IN MXFL              CGTM        645     A
CGTM              MX7    44                                                           CGTM        646     A
CGTM              ZR     X6,MOREFL   IF WA = 0 */ MAX FL USED ALREADY                 CGTM        647     A
CGTM              SB3    X6                                                           CGTM        648     A
CGTM              MX7    -6                                                           CGTM        649     A
CGTM              IX2    X1-X7                                                        CGTM        650     A
CGTM              BX2    X7*X2       ROUND WN TO NEXT 100B                            CGTM        651     A
CGTM              SB2    X2+1000B                                                     CGTM        652     A
CGTM              GE     B2,B3,MFL1                                                   CGTM        653     A
CGTM              SX6    B2          WR = MIN(WA,WN+1000B) */ NO OF WDS REQ.          CGTM        654     A
CGTM                                                                                  CGTM        655     A
CGTM      MFL1    IX7    X6+X3                                                        CGTM        656     A
CGTM              LX7    30                                                           CGTM        657     A
CGTM              SA7    MFLA                                                         CGTM        658     A
CGTM              BX7    X1                                                           CGTM        659     A
CGTM              SA7    A7+B1                                                        CGTM        660     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM              PRINT  MOREFL,(* WA,WR =*2Z7),(B3,X6)                    CGTM        661     A
CGTM              MEMORY SCM,MFLA,RCL                                      CGTM        662     A
CGTM              SA2    MFLA                                              CGTM        663     A
CGTM              SA3    CP.AFLS     OFL = OLD FL                          CGTM        664     A
CGTM              AX2    30          NFL = NEW FL                          CGTM        665     A
CGTM              IX0    X2-X3       WG = NO. OF WORDS GRANTED             CGTM        666     A
CGTM              BX6    X2                                                CGTM        667     A
CGTM              SA6    A3          CP.AFLS = NEW FL                      CGTM        668     A
CGTM              SA3    =XHO$PMLS                                         CGTM        669     A
CGTM              MX7    X2+X3       HO$PMLS = MAX( HO$PMLS , NFL )        CGTM        670     A
CGTM              SA7    A3                                                CGTM        671     A
CGTM              PRINT  MOREFL,(*  NEW FL,WG =*2Z7),(X6,X0)               CGTM        672     A
CGTM              RJ     AST         */ ADJUST SPECIAL TABLES              CGTM        673     A
CGTM              SA1    MFLB                                              CGTM        674     A
CGTM              BX6    X0          (X6) = NO.OF WORDS GRANTED */ EXIT PAR. CGTM      675     A
CGTM              IX7    X0-X1                                             CGTM        676     A
CGTM              SA2    THRESH                                            CCGA025       5     A
CGTM              PL     X7,MOREFL   IF ENOUGH SPACE GRANTED               CCGA025       6     A
CGTM              MX7    0                                                 CCGA025       7     A
CGTM              IX6    X6+X2                                             CCGA025       8     A
CGTM              SA7    A2          RELINQUISH TABLE MANAGER SLOP         CCGA025       9     A
CGTM              IX7    X6-X1                                             CCGA025      10     A
CGTM              EQ     MOREFL                                            CGTM        677     A
CGTM                                                                      CGTM        678     A
CGTM       MFLA   BSS    1                                                 CGTM        679     A
CGTM       MFLB   BSS    1                                                 CGTM        680     A
CGTM       CWS=   SPACE  3,14                                              CGTM        681     A
CGTM      **      CWS# - CALCULATE WORKING STORAGE SIZE                    CGTM        682     A
CGTM      *                                                               CGTM        683     A
CGTM      *       EXIT   (X6) = CURRENT WORKING STORAGE SIZE               CGTM        684     A
CGTM      *              (X7) = MAXIMUM WORKING STORAGE SIZE               CGTM        685     A
CGTM                                                                      CGTM        686     A
CGTM       CWS    ENTRY. **,#                                             CGTM        687     A
CGTM              SA1    LM                                                CGTM        688     A
CGTM              SA2    FTAB+N.DT+HC.NDAT LWA OF WS AREA                  CGTM        689         I
          -CCG29
CGTM              SA2    FTAB+N.DT   LWA OF DYNAMIC TABLES                 CCG29         6     A
CGTM              SA3    =XCP.AFLS                                         CGTM        690     A
CGTM              SA4    =XHO$MFLS                                         CGTM        691     A
CGTM              AX4    30                                                CGTM        692     A
CGTM              IX6    X2-X1       CWS = O.TEND - LM                     CGTM        693     A
CGTM              IX7    X4-X3                                             CGTM        694     A
CGTM              IX7    X7+X6       MWS = MAX.FL - CP.AFLS + CWS          CGTM        695     A
CGTM              EQ     CWS                                               CGTM        696     A
CGTM                                                                      CGTM        697     A
CGTM                                                                      CGTM        698     A
CGTM      **      AST - ADJUST SPECIAL TABLES IN HIGH CORE AND POINTERS TO THEM  CGTM  699     A
CGTM      *                                                               CGTM        700     A
CGTM      *       ENTRY  (X0) = CHANGE                                    CGTM        701     A
CGTM                                                                      CGTM        702     A
CGTM       AST    ROUTINE                                                 CGTM        703     A
CGTM              PRINT  AST,(* AST CALLED,CHANGE =*Z7),(X0)               CGTM        704     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM              SA4    TN                                                    CGTM      705    A
CGTM              SA2    FTAB-1+X4   FWA OF STATIC TABLES                      CGTM      706    A
CGTM              SB2    X4-1                                                  CGTM      707    A
CGTM              SA4    =XCP.NFLS   FL-10                                     CGTM      708    A
CGTM              IX7    X4+X0       NEW CP.NFLS = OLD CP.NFLS + CHANGE        CGTM      709    A
CGTM              SA7    A4                                                    CGTM      710    A
CGTM              IX1    X4-X2       WDS = CP.NFLS - O.TEND                    CGTM      711    A
CGTM              IX3    X2+X0       TO = O.TEND + CHANGE                      CGTM      712    A
CGTM              SB3    NTAB                                                  CGTM      713    A
CGTM      AST1    SA4    FTAB+B2                                               CGTM      714    A
CGTM              IX7    X4+X0                                                 CGTM      715    A
CGTM              SB2    B2+B1                                                 CGTM      716    A
CGTM              SA7    A4                                                    CGTM      717    A
CGTM              LT     B2,B3,AST1                                            CGTM      718    A
CGTM              MOVE   X1,X2,X3                                              CGTM      719    A
CGTM              EQ     AST                                                   CGTM      720    A
CGTM      CG$PTC  TITLE  CG$PTC - PRINT *CCG* TABLES AFTER A REPRIEVE          CGTM      721    A
CGTM      .T      IFNE   TEST,0                                 * TEST MODE *  CGTM      722    A
CGTM                                                                           CGTM      723    A
CGTM      NAMTAB  BSS    0                                                     CGTM      724    A
CGTM      TABNAM  HERE                                                         CGTM      725    A
CGTM                                                                           CGTM      726    A
CGTM      CG$PTC  ENTRY.                                                       CGTM      727    A
CGTM              SETB1                                                        CGTM      728    A
CGTM              PRINT  ,(//*  -- COMPILER TABLE DUMP --*/)                   CGTM      729    A
CGTM              SX1    4           SET TO *EXECUTIVE IS /PTC=/*              CGTM      730    A
CGTM              RJ     =XSOB       SET OUTPUT BIT FLAGS                      CGTM      731    A
CGTM              SX6    1                                                     CGTM      732    A
CGTM              SA6    PTCA        INITIALIZE COUNTER                        CGTM      733    A
CGTM                                                                           CGTM      734    A
CGTM      PTC1    SA1    PTCA                                                  CGTM      735    A
CGTM              SA2    FTAB+X1     GET FWA OF TABLE                          CGTM      736    A
CGTM              SA3    LTAB+X1     GET LENGTH OF TABLE                       CGTM      737    A
CGTM              SA5    NAMTAB+X1   GET NAME OF TABLE                         CGTM      738    A
CGTM              RJ     PMT                                                   CGTM      739    A
CGTM              SA1    PTCA                                                  CGTM      740    A
CGTM              SA2    TN                                                    CGTM      741    A
CGTM              SX6    X1+B1                                                 CGTM      742    A
CGTM              SA6    A1          INCREMENT COUNTER                         CGTM      743    A
CGTM              IX7    X6-X2                                                 CGTM      744    A
CGTM              MI     X7,PTC1     IF MORE TABLES                            CGTM      745    A
CGTM                                                                           CGTM      746    A
CGTM              SA1    =XHO$OPT                                              CGTM      747    A
CGTM              PL     X1,PTC2     IF OPT " 2                                CGTM      748    A
CGTM              ECHO   4,TBL=(BIT,SEQ)                                       CGTM      749    A
CGTM              SA2    =XO$TBL                                               CGTM      750    A
CGTM              SA3    =XL$TBL                                               CGTM      751    A
CGTM              SA5    =10H TBL                                              CGTM      752    A
CGTM              RJ     PMT                                                   CGTM      753    A
CGTM                                                                           CGTM      754    A
CGTM      PTC2    SA1    L$TXT                                                 CGTM      755    A
CGTM              SB2    X1-8                                                  CGTM      756    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM                LE     B2,PTC3                                                  CGTM        757     A
CGTM                SA1    O$TXT                                                    CGTM        758     A
CGTM                SA2    X1                                                       CGTM        759     A
CGTM                UX6    B2,X2                                                    CGTM        760     A
CGTM                LE     B2,PTC3     IF NO EXPONENT                               CGTM        761     A
CGTM                SNAPRL PTC                                                      CGTM        762     A
CGTM                                                                                CGTM        763     A
CGTM       PTC3     RECALL =XF.OUT                                                  CGTM        764     A
CGTM                EQ     CG$PTC                                                   CGTM        765     A
CGTM                                                                                CGTM        766     A
CGTM       PTCA     BSS    1                                                        CGTM        767     A
CGTM                                                                                CGTM        768     A
CGTM       PMT      ROUTINE                                                         CGTM        769     A
CGTM                ZR     X3,PMT      IF LEN = 0                                   CGTM        770     A
CGTM                PRINT  ,(/*  TABLE -*A10*  FWA,LEN =*2Z6/),(X5,X2,X3)          CGTM        771     A
CGTM                IX3    X2+X3                                                    CGTM        772     A
CGTM                MX4    0                                                        CGTM        773     A
CGTM                RJ     =XDCM=      DUMP CENTRAL MEMORY                          CGTM        774     A
CGTM                EQ     PMT                                                      CGTM        775     A
CGTM                                                                                CGTM        776     A
CGTM       .T       ENDIF                                          * TEST MODE *    CGTM        777     A
CGTM       ATS      TITLE  TABLE MANAGER SUBROUTINES                                CGTM        778     A
CGTM     **         ATS$ - ALLOCATE TABLE SPACE                                     CGTM        779     A
CGTM     *                                                                          CGTM        780     A
CGTM     *          ENTRY  (A0) = TABLE INDEX.                                      CGTM        781     A
CGTM     *                 (X1) = CHANGE (+ OR -) TO TABLE SIZE.                    CGTM        782     A
CGTM     *                                                                          CGTM        783     A
CGTM     *          EXIT   (X1) = CHANGE.                                           CGTM        784     A
CGTM     *                 (X2) = FWA OF TABLE.                                     CGTM        785     A
CGTM     *                 (X3) = NEW LENGTH OF TABLE.                              CGTM        786     A
CGTM     *                 (B6) = PREVIOUS TABLE LENGTH                             CGTM        787     A
CGTM     *                                                                          CGTM        788     A
CGTM     *          USES   X - 0, 1, 2, 3, 4, 5, 6, 7.                              CGTM        789     A
CGTM     *                 B - 2, 3, 4, 5, 7                                        CGTM        790     A
CGTM     *                 A - 1, 2, 3, 4, 6, 7                                     CGTM        791     A
CGTM     *          RESTORES X0, X5.                                                CGTM        792         I
         -CCG15
CGTM     *          RESTORES X0, X5, B4, B5, B7                                     CCG15       702     A
CGTM     *          CALLS  AMU, MVE.                                                CGTM        793     A
CGTM     *          THE DYNAMIC TABLES ARE DIVIDED INTO TWO PORTIONS.  THE LOW      CCGA011      19     A
CGTM     *          PORTION RECIEVES 7/8 OF THE FREE SPACE AVAILABLE, THEREFORE     CCGA011      20     A
CGTM     *          THE LARGER AND FAST GROWING TABLES ARE IN THE LOW PORTION.      CCGA011      21     A
CGTM     *          ATSNAT# IS THE NUMBER OF ACTIVE TABLES (IN LOW).                CCGA011      22     A
CGTM     *          ALSO, 6000B OF FREE STORAGE IS ALWAYS KEPT IN RESERVE.          CCGA011      23     A
CGTM     *          OPT = 1, ATSNAT# = 7.                                           CCGA011      24     A
CGTM     *          OPT = 2, BEFORE AFT IS CALLED.  ATSNAT# = 8.                    CCGA011      25     A
CGTM     *          OPT = 2, AFTER AFT IS CALLED.   ATSNAT# = 6.                    CCGA011      26     A
CGTM                                                                                CCGA011      27     A
CGTM                ENTRY  ATSNAT#,ATSTOG#                                          CCGA011      28     A
CGTM       ATSNAT#  CON    8           NO. OF ACTIVE TABLES  (IN LOW)              CCGA011      29     A
CGTM       ATSTOG#  CON    6&8         ATSNAT TOGGLE                               CCGA011      30     A
CGTM                                                                                CGTM        794     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM      ATSX     SA2    FTAB+A0     (X2) = FWA TABLE                                CGTM        795     A
CGTM               SA3    LTAB+A0     (X3) = NEW LENGTH                               CGTM        796     A
CGTM                                                                                  CGTM        797     A
CGTM      ATS      ENTRY. **,$                                                        CGTM        798     A
CGTM      ATS1     SA2    FTAB+A0     CURRENT FWA                                     CGTM        799     A
CGTM               SA3    LTAB+A0     CURRENT LENGTH                                  CGTM        800     A
CGTM               SA4    A2+B1       NEXT TABLE FWA                                  CGTM        801     A
CGTM               IX6    X1+X3       NEW LENGTH                                      CGTM        802     A
CGTM               SB6    X3          (B6) = PREVIOUS TABLE LENGTH                    CGTM        803     A
CGTM               IX7    X4-X2       ROOM BETWEEN TABLES                             CGTM        804     A
CGTM               SA6    A3          SET NEW LENGTH                                  CGTM        805     A
CGTM               IX7    X7-X6                                                       CGTM        806     A
CGTM               PL     X7,ATSX     RETURN IF ROOM FOR CHANGE                       CGTM        807     A
CGTM               SX7    X3          SET PREVIOUS LENGTH                             CGTM        808     A
CGTM               PRINT  ATS,(* CRASHER, WORDS NEEDED = *2Z7),(A0,X1)                CCGA011      31     A
CGTM                                                                                  CGTM        809     A
CGTM     **        INITIALIZE FOR TABLE MOVE.                                         CGTM        810     A
CGTM                                                                                  CGTM        811     A
CGTM               SA2    TN          (B2) = NUMBER OF TABLES                         CGTM        812     A
CGTM               BX6    X0          SAVE (X0)                                       CGTM        813     A
CGTM               SA7    A3                                                          CGTM        814     A
CGTM               SA7    PL          PL = PREVIOUS LENGTH                            CGTM        815     A
CGTM               SB2    X2                                                          CGTM        816     A
CGTM               LX7    X5          SAVE (X5)                                       CGTM        817     A
CGTM               SA6    A2+B1                                                       CGTM        818     A
CGTM               SA7    A6+B1                                                       CGTM        819     A
CGTM               BX6    X1          SAVE (X1)                                       CGTM        820     A
CGTM               SA6    A7+B1                                                       CGTM        821     A
CGTM               MX6    -18         SAVE B4, B5, B7                                 CCG15       703     A
CGTM               SX7    B4                                                          CCG15       704     A
CGTM               SX2    B5                                                          CCG15       705     A
CGTM               BX7    -X6*X7                                                      CCG15       706     A
CGTM               BX2    -X6*X2                                                      CCG15       707     A
CGTM               LX7    18                                                          CCG15       708     A
CGTM               BX2    X7+X2                                                       CCG15       709     A
CGTM               SX7    B7                                                          CCG15       710     A
CGTM               BX7    -X6*X7                                                      CCG15       711     A
CGTM               LX2    18                                                          CCG15       712     A
CGTM               BX7    X7+X2                                                       CCG15       713     A
CGTM               SA7    ATSA                                                        CCG15       714     A
CGTM                                                                                  CGTM        822     A
CGTM     **        COMPUTE REMAINING TABLE SPACE.                                     CGTM        823     A
CGTM                                                                                  CGTM        824     A
CGTM               SB3    B2-B1                                                       CGTM        825     A
CGTM               BX4    X1          INCLUDE INCREASE                                CGTM        826         I
          -CCGA011
CGTM      ATS2     SB3    B3-B1       ACCUMULATE ASSIGNED TABLE LENGTHS               CGTM        827         I
          -CCGA011
CGTM      ATS5     SA2    ATSNAT#                                                     CCGA011      32     A
CGTM               SB5    X2-1        LSTLOW = ATSNAT-1                               CCGA011      33     A
CGTM               SX4    B0                                                          CCGA011      34     A
CGTM      ATS10    SB3    B3-B1       ACCUMULATE ASSIGNED TABLE LENGTHS               CCGA011      35     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM              SA3    LTAB+B3                                          CGTM        828     A
CGTM              IX4    X4+X3                                            CGTM        829         I
         -CCGA011
CGTM              NZ     B3,ATS2     LOOP FOR ALL TABLES                  CGTM        830         I
         -CCGA011
CGTM              NE     B3,B5,ATS15 IF I " LSTLOW                        CCGA011      36     A
CGTM              SB6    X4          SAVE HIGHAL                          CCGA011      37     A
CGTM      ATS15   IX4    X4+X3                                            CCGA011      38     A
CGTM              NZ     B3,ATS10    LOOP FOR ALL TABLES                  CCGA011      39     A
CGTM              IX4    X4+X1       TOTAL = TOTAL + CHANGE               CCGA011      40     A
CGTM              SB4    A0-B5                                            CCGA011      41     A
CGTM              LE     B4,B0,ATS20 IF CRASHER LE LSTLOW                 CCGA011      42     A
CGTM              SB6    X1+B6       HIGHAL = HIGHAL + CHANGE             CCGA011      43     A
CGTM      ATS20   SX7    B6                                               CCGA011      44     A
CGTM              SA7    ATSB                                             CCGA011      45     A
CGTM              SA2    LM          SET AVAILABLE LENGTH                 CGTM        831     A
CGTM              SA3    FTAB-1+B2                                        CGTM        832     A
CGTM              IX6    X3-X2                                            CGTM        833     A
CGTM              SB4    X4          (B4) = TOTAL ASSIGNED LENGTH         CGTM        834     A
CGTM              IX7    X6-X4                                            CGTM        835     A
CGTM              SB5    X7          (B5) = REMAINING SPACE               CGTM        836     A
CGTM              NG     X7,ATS8     IF NO REMAINING SPACE                CGTM        837         I
         -CCGA011
CGTM              SA1    THRESH      1000B/6000B OPT = 1/2                CCGA011      46     A
CGTM              IX7    X7-X1       GUARANTEE ELBOW ROOM                 CCGA011      47     A
CGTM              MI     X7,ATS60    IF NO REMAINING SPACE                CCGA011      48     A
CGTM                                                                     CGTM        838     A
CGTM     **       MOVE ALL TABLES DOWN.                                  CGTM        839     A
CGTM                                                                     CGTM        840     A
CGTM              SA1    LM          LOW MEMORY LIMIT                     CGTM        841     A
CGTM              SB3    B1                                               CGTM        842     A
CGTM              LX0    X1                                               CGTM        843     A
CGTM      ATS3    SA2    FTAB-1+B3   ORIGIN = PREVIOUS FWA                CGTM        844         I
         -CCGA011
CGTM      ATS25   SA2    FTAB-1+B3   ORIGIN = PREVIOUS FWA                CCGA011      49     A
CGTM              SA1    LTAB-1+B3   WORD COUNT = LENGTH                  CGTM        845     A
CGTM              BX3    X0          DESTINATION = NEW FWA                CGTM        846     A
CGTM              IX0    X0+X1       NEXT FWA                             CGTM        847     A
CGTM              SX7    X3          SET NEW FWA                          CGTM        848     A
CGTM              BX4    X2-X3       MOVE DIFFERENTIAL                    CGTM        849     A
CGTM              SA7    A2                                               CGTM        850     A
CGTM              SB3    B3+B1       COUNT TABLE                          CGTM        851     A
CGTM              ZR     X4,ATS4     IF NO MOVE REQUIRED                  CGTM        852         I
         -CCGA011
CGTM              ZR     X4,ATS30    IF NO MOVE REQUIRED                  CCGA011      50     A
CGTM              RJ     MVE=                                             CGTM        853     A
CGTM      ATS4    NE     B3,B2,ATS3  LOOP FOR ALL TABLES                  CGTM        854         I
         -CCGA011
CGTM      ATS30   NE     B3,B2,ATS25 LOOP FOR ALL TABLES                  CCGA011      51     A
CGTM              SA1    TN+3        INCREMENT SIZE OF REQUESTED TABLE    CGTM        855     A
CGTM              SA2    LTAB+A0                                          CGTM        856     A
CGTM              IX6    X2+X1                                            CGTM        857     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM                SA6    A2                                                   CGTM        858     A
CGTM                                                                            CGTM        859     A
CGTM      **        REALLOCATE TABLES.                                           CGTM        860     A
CGTM      *         ALLOCATE INTERSPACE.  (LA/2N) + ((TL*LA)/2)                  CGTM        861         I
          -CCGA011
CGTM      *         ALLOCATE INTERSPACE FOR HIGH. 1/8*(HIGHLA/2)*(1/N+TL/HIGHAL) CCGA011      52     A
CGTM                                                                            CGTM        862     A
CGTM                SA1    FTAB-1+B2   (X0) = LWA+1 LAST TABLE                   CGTM        863     A
CGTM                SB3    B2-B1                                                CGTM        864     A
CGTM                BX0    X1                                                   CGTM        865     A
CGTM       ATS4A    SB6    2           LOWER LIMIT = 2 ( FIRST TABLE INACTIVE ) CGTM        866         I
          -CCGA011
CGTM                SB2    B2-1        N = N - 1                                CGTM        867         I
          -CCGA011
CGTM       ATS5     SB3    B3-B1       I = I + 1                                CGTM        868         I
          -CCGA011
CGTM                SA2    ATSNAT#                                              CCGA011      53     A
CGTM                SB6    X2          LC = ATSNAT                              CCGA011      54     A
CGTM                SB2    B2-B6       NTABS = TOTAL TABS - ATSNAT              CCGA011      55     A
CGTM                SX7    B4                                                   CCGA011      56     A
CGTM                SA7    ATSC        SAVE TOTAL                               CCGA011      57     A
CGTM                SX6    B5                                                   CCGA011      58     A
CGTM                SA6    ATSD        SAVE TOTLA                               CCGA011      59     A
CGTM                SA2    ATSB                                                 CCGA011      60     A
CGTM                SB4    X2          AL = HIGHAL                              CCGA011      61     A
CGTM                AX6    3           HIGHLA = TOTLA/8                         CCGA011      62     A
CGTM                SB5    X6          LA = HIGHLA                              CCGA011      63     A
CGTM                                                                            CCGA011      64     A
CGTM      *         DO ACTUAL REALOCATION.                                       CCGA011      65     A
CGTM                                                                            CCGA011      66     A
CGTM       ATS35    SB3    B3-B1       I = I + 1                                CCGA011      67     A
CGTM                SA2    FTAB+B3     FWA(I)                                   CGTM        869     A
CGTM                SX4    B5          LA (LENGTH AVAILABLE)                    CGTM        870     A
CGTM                SA1    LTAB+B3     TL (LENGTH OF TABLE)                     CGTM        871     A
CGTM                SX3    B2-B1       N  (NUMBER OF TABLES)                    CGTM        872     A
CGTM                AX5    X4,B1       LA/2                                     CGTM        873     A
CGTM                IX6    X4/X3       L1 = LA/N                                CGTM        874     A
CGTM                ZR     B4,ATS6     IF NO TABLES ASSIGNED, L = L2            CGTM        875         I
          -CCGA011
CGTM                ZR     B4,ATS40                                             CCGA011      68     A
CGTM                SX3    B4          AL (TOTAL ASSIGNED LENGTH)               CGTM        876     A
CGTM                IX7    X5*X1       (LA/2)*TL                                CGTM        877     A
CGTM                AX6    1           L1 = LA/2N                               CGTM        878     A
CGTM                IX7    X7/X3       L2 = (TL*LA)/2                           CGTM        879     A
CGTM                IX6    X6+X7       L = L1+L2                                CGTM        880     A
CGTM                SX1    X1                                                   CGTM        881     A
CGTM       ATS6     IX6    X0-X6       FWA(I+1)-L                               CGTM        882         I
          -CCGA011
CGTM       ATS40    IX6    X0-X6       FWA(I+1) - L                             CCGA011      69     A
CGTM                IX7    X6-X1       FWA(I) = FWA(I+1)-L-TL                   CGTM        883     A
CGTM                BX3    X7          DESTINATION = FWA(I)                     CGTM        884     A
CGTM                SA7    A2                                                   CGTM        885     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM              LX0    X7          FWA(I+1) = FWA(I)                          CGTM        886     A
CGTM              BX4    X2-X3       CHECK MOVE DIFFERENTIAL                    CGTM        887     A
CGTM              ZR     X4,ATS7     IF NO MOVE REQUIRED                        CGTM        888         I
          -CCGA011
CGTM              ZR     X4,ATS45    IF NO MOVE REQUIRED                        CCGA011      70     A
CGTM              RJ     MVE=        MOVE TABLE                                 CGTM        889     A
CGTM       ATS7   GT     B3,B6,ATS5  LOOP                                       CGTM        890         I
          -CCGA011
CGTM       ATS45  GT     B3,B6,ATS35 LOOP                                       CCGA011      71     A
CGTM                                                                            CCGA011      72     A
CGTM      *       ALLOCATE INTERSPACE FOR LOW. 7/8*(LOWLA/2)*(1/N+TL/LOWAL)     CCGA011      73     A
CGTM                                                                            CCGA011      74     A
CGTM              SX1    B6-3                                                   CCGA011      75     A
CGTM              MI     X1,ATS55   IF LC < 3    */ALREADY DID LOW              CCGA011      76     A
CGTM              SA2    ATSFAT                                                 CCGA011      77     A
CGTM              SB6    X2          FIRST ACTIVE TABLE                         CCGA011      78     A
CGTM      +       SA2    ATSNAT#                                                CCGA011      79     A
CGTM              SB2    X2+B1                                                  CCGA011      80     A
CGTM              SB2    B2-B6       NTABS = ATSNAT + 1 - LC                    CCGA011      81     A
CGTM              SA1    ATSC                                                   CCGA011      82     A
CGTM              SX2    B4                                                     CCGA011      83     A
CGTM              IX3    X1-X2       LOWAL = TOTAL - HIGHAL                     CCGA011      84     A
CGTM              SB4    X3          AL = LOWAL                                 CCGA011      85     A
CGTM              SA1    ATSD                                                   CCGA011      86     A
CGTM              SX2    B5                                                     CCGA011      87     A
CGTM              IX3    X1-X2       LOWLA = TOTLA - HIGHLA                     CCGA011      88     A
CGTM              SB5    X3          LA = LOWLA                                 CCGA011      89     A
CGTM              EQ     ATS35                                                  CCGA011      90     A
CGTM                                                                            CGTM        891     A
CGTM      **      RESTORE REGISTERS.                                            CGTM        892     A
CGTM                                                                            CGTM        893     A
CGTM              SA1    ATSA        COUNT MOVES                                CGTM        894         I
          -CCG15
CGTM              SX6    X1+B1                                                  CGTM        895         I
          -CCG15
CGTM              SA6    A1                                                     CGTM        896         I
          -CCG15
CGTM              SA1    ATSA        RESTORE B-REGS                             CCG15       715         I
          -CCGA011
CGTM       ATS55  SA1    ATSA                                                   CCGA011      91     A
CGTM              SB7    X1                                                     CCG15       716     A
CGTM              AX1    18                                                     CCG15       717     A
CGTM              SB5    X1                                                     CCG15       718     A
CGTM              AX1    18                                                     CCG15       719     A
CGTM              SB4    X1                                                     CCG15       720     A
CGTM              SA2    TN+1        RESTORE (X0)                               CGTM        897     A
CGTM              SA3    A2+B1       RESTORE (X5)                               CGTM        898     A
CGTM              BX0    X2                                                     CGTM        899     A
CGTM              SA4    PL                                                     CGTM        900     A
CGTM              LX5    X3                                                     CGTM        901     A
CGTM              SA1    A3+B1       RESTORE (X1)                               CGTM        902     A
CGTM              SB6    X4          (B6) = PREVIOUS LENGTH OF TABLE            CGTM        903     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM                EQ     ATSX        RETURN                                          CGTM         904     A
CGTM                                                                                   CGTM         905     A
CGTM      ATS8      SA3    TO          PROCESS TABLE OVERFLOW                          CGTM         906         I
         -CCGA011
CGTM      ATS60     SA3    TO          PROCESS TABLE OVERFLOW                          CCGA011       92     A
CGTM                SB5    X7                                                          CCGA011       93     A
CGTM                SX6    A0                                                          CGTM         907     A
CGTM                SB2    X3                                                          CGTM         908     A
CGTM                SA6    TN+4        (A0)                                            CGTM         909     A
CGTM                JP     B2          (B5) = - WORDS NEEDED                           CGTM         910     A
CGTM                                                                                   CGTM         911     A
CGTM     *          RETURN FROM USER TABLE OVERFLOW PROCESSOR ( SUCCESS )              CGTM         912     A
CGTM                                                                                   CGTM         913     A
CGTM      ATS9      SA1    TN+3        X1                                              CGTM         914         I
         -CCGA011
CGTM      ATS65     SA1    TN+3        X1                                              CCGA011       94     A
CGTM                SA4    A1-B1       X5                                              CGTM         915     A
CGTM                BX5    X4                                                          CGTM         916     A
CGTM                SA3    A1+B1       A0                                              CGTM         917     A
CGTM                SA0    X3                                                          CGTM         918     A
CGTM                SA4    A4-B1       X0                                              CGTM         919     A
CGTM                BX0    X4                                                          CGTM         920     A
CGTM                SA2    A4-B1       TN                                              CGTM         921     A
CGTM                SB2    X2          (B2) = TN                                       CGTM         922     A
CGTM                SB3    B2-B1                                                       CGTM         923     A
CGTM                BX4    X1                                                          CGTM         924         I
         -CCGA011
CGTM                EQ     ATS2                                                        CGTM         925         I
         -CCGA011
CGTM                EQ     ATS5                                                        CCGA011       95     A
CGTM                                                                                   CGTM         926     A
CGTM      ATSA      CON    0           COUNT OF TABLE MOVES                            CGTM         927         I
         -CCG15
CGTM      ATSA      CON    0           B4,B5, B7 SAVE IF A TABLE MOVE                  CCG15        721     A
CGTM      ATSB      BSS    1                                                           CCGA011       96     A
CGTM      ATSC      BSS    1                                                           CCGA011       97     A
CGTM      ATSD      BSS    1                                                           CCGA011       98     A
CGTM      ATSFAT    CON    2           FIRST ACTIVE TABLE                              CCGA011       99     A
CGTM      AFT       SPACE  3,14                                                        CGTM         928     A
CGTM     **         AFT - ACTIVATE FIRST TABLE                                         CGTM         929     A
CGTM                                                                                   CGTM         930     A
CGTM      AFT       ENTRY. **,#                                                        CGTM         931     A
CGTM                SA1    AFTA                                                        CGTM         932         I
         -CCGA011
CGTM                SA2    LM                                                          CGTM         933         I
         -CCGA011
CGTM                SA3    ATS4A                                                       CGTM         934         I
         -CCGA011
CGTM                BX6    X1                                                          CGTM         935         I
         -CCGA011
CGTM                LX7    X2                                                          CGTM         936         I
         -CCGA011

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM                SA6    ATS4A       STORE PLUG                                  CGTM         937          I
            -CCGA011
CGTM                SA7    FTAB                                                    CGTM         938          I
            -CCGA011
CGTM                BX7    X3                                                      CGTM         939          I
            -CCGA011
CGTM                SA7    A1                                                      CGTM         940          I
            -CCGA011
CGTM                EQ     AFT                                                     CGTM         941          I
            -CCGA011
CGTM                                                                               CGTM         942          I
            -CCGA011
CGTM         AFTA   SB6    1                                                       CGTM         943          I
            -CCGA011
CGTM                SB0    0                                                       CGTM         944          I
            -CCGA011
CGTM                SA2    ATSFAT                                                  CCGA011      100     A
CGTM                SX1    1&2                                                     CCGA011      101     A
CGTM                BX6    X2-X1                                                   CCGA011      102     A
CGTM                SA6    A2          TOGGLE ATSFAT                               CCGA011      103     A
CGTM                SA1    ATSNAT#                                                 CCGA011      104     A
CGTM                SA2    ATSTOG#                                                 CCGA011      105     A
CGTM                BX6    X2-X1                                                   CCGA011      106     A
CGTM                SA6    A1          TOGGLE ATSNAT                               CCGA011      107     A
CGTM                SA1    LM                                                      CCGA011      108     A
CGTM                BX7    X1                                                      CCGA011      109     A
CGTM                SA7    FTAB                                                    CCGA011      110     A
CGTM                EQ     AFT                                                     CCGA011      111     A
CGTM         ADW    EJECT              ADW                                         CGTM         945     A
CGTM        **      ADW$ - ADD A WORD TO THE END OF A MANAGED TABLE                CGTM         946     A
CGTM        *       ENTRY  (A0) = TABLE NUMBER.                                    CGTM         947     A
CGTM        *              (X1) = WORD.                                            CGTM         948     A
CGTM        *       EXIT   (X1) = WORD.                                            CGTM         949     A
CGTM        *              (X6) = WORD.                                            CGTM         950     A
CGTM        *              (A6) = ADDRESS OF WORD.                                 CGTM         951     A
CGTM        *              (X2) = FWA TABLE.                                       CGTM         952     A
CGTM        *              (X3) = LENGTH OF TABLE.                                 CGTM         953     A
CGTM        *              (B6) = L.TBL - 1                                        CGTM         954     A
CGTM        *                                                                      CGTM         955     A
CGTM        *       USES   X - 1, 6, 7.                                            CGTM         956     A
CGTM        *              B - NONE.                                               CGTM         957     A
CGTM        *              A - 1, 6.                                               CGTM         958     A
CGTM        *       CALLS  ATS.                                                    CGTM         959     A
CGTM                                                                               CGTM         960     A
CGTM         ADW1   BX7    X3                                                      CGTM         961     A
CGTM                IX4    X2+X3                                                   CGTM         962     A
CGTM                SA7    A3+         UPDATE LENGTH                               CGTM         963     A
CGTM                SB6    X3-1              (B6) = OLD LENGTH                     CGTM         964     A
CGTM                SA6    X4-1        STORE WORD                                  CGTM         965     A
CGTM         ADW    ENTRY. **,$                                                    CGTM         966     A
CGTM                SA2    FTAB+A0     FWA                                         CGTM         967     A
CGTM                SA3    LTAB+A0                                                 CGTM         968     A

            0         1         2         3         4         5         6         7         8
            12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

```
CGTM              SA4    A2+B1       NEXT TABLE FWA                          CGTM        969     A
CGTM              BX6    X1                                                  CGTM        970     A
CGTM              IX7    X4-X2       ROOM BETWEEN TABLES                     CGTM        971     A
CGTM              SX3    X3+B1                                               CGTM        972     A
CGTM              IX4    X7-X3                                               CGTM        973     A
CGTM              PL     X4,ADW1     IF ROOM TO ADD A WORD                   CGTM        974     A
CGTM              SX1    B1                                                  CGTM        975     A
CGTM              SA6    ADWA                                                CGTM        976     A
CGTM              RJ     ATS         ALLOCATE SPACE                          CGTM        977     A
CGTM              SA1    ADWA                                                CGTM        978     A
CGTM              IX4    X2+X3                                               CGTM        979     A
CGTM              BX6    X1                                                  CGTM        980     A
CGTM              SA6    X4-1                                                CGTM        981     A
CGTM              EQ     ADW                                                 CGTM        982     A
CGTM                                                                        CGTM        983     A
CGTM       ADWA   BSS    1                                                   CGTM        984     A
CGTM       MTU    SPACE  3                                                   CGTM        985     A
CGTM      **      MTU$ - MOVE TABLES UP                                      CGTM        986     A
CGTM      *       ENTRY  NONE.                                               CGTM        987     A
CGTM      *       EXIT   NONE.                                               CGTM        988     A
CGTM      *       USES   X - 0, 1, 2, 3, 7.                                  CGTM        989     A
CGTM      *              B - 3.                                              CGTM        990     A
CGTM      *              A - 1, 2, 7.                                        CGTM        991     A
CGTM      *       CALLS  MVE.                                                CGTM        992     A
CGTM                                                                        CGTM        993     A
CGTM       MTU1   SB3    B3-B1       DECREMENT TABLE COUNT                   CGTM        994     A
CGTM              SA2    FTAB+B3                                             CGTM        995     A
CGTM              SA1    LTAB+B3                                             CGTM        996     A
CGTM              IX7    X0-X1       NEW FWA = L - LENGTH                    CGTM        997     A
CGTM              LX0    X7          L = NEW FWA                             CGTM        998     A
CGTM              SA7    A2                                                  CGTM        999     A
CGTM              BX3    X0          MOVE TABLE                              CGTM       1000     A
CGTM              ZR     X1,MTU2     IF LEN = 0                              CGTM       1001     A
CGTM              IX4    X2-X3                                               CGTM       1002     A
CGTM              ZR     X4,MTU2                                             CGTM       1003     A
CGTM              MOVE   X1,X2,X3                                            CGTM       1004     A
CGTM       MTU2   NZ     B3,MTU1                                             CGTM       1005     A
CGTM                                                                        CGTM       1006     A
CGTM       MTU    ENTRY. **,$                                               CGTM       1007     A
CGTM              SB3    NTAB-1      (B3) = NUMBER OF TABLES - 1             CGTM       1008     A
CGTM              SA1    FTAB+B3     (X0) = LWA+1 OF ALL TABLES (L)          CGTM       1009     A
CGTM              BX0    X1                                                  CGTM       1010     A
CGTM              EQ     MTU1                                                CGTM       1011     A
CGTM                                                                        CGTM       1012     A
CGTM              END                                                         CGTM       1013     A
```

SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - CGTM

IDENTIFIER	TOTAL	ACTIVE
CGTM	1013	810
CCG15	721	654

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CGTM

CCG16                    136       129
CCG18                     70        69
CCG*78                     4         2
CCG24                      1         1
CCG27                      1         1
CCG29                      1         1
CCG30                      1         1
CCG35                      6         6
CCG38                     12        12
CCGA011                   99        99
CCGA018                    3         3
CCGA021                   20        20
CCG*80                     1         0
CCGA025                    6         6
CCGA027                    5         5
CCGA028                    1         1
CCG*81                     1         0
CCGA037                    1         1
CCGA041                    2         2
CCG*82                     1         0
CCGA072                   17        17
CCGA092                    3         3
CCG0187                    1         1
CCG0220                    2         2
CCG1*92                    1         0
CCG*94                     1         1

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OUTPTK

      OUTPTK   *DECK     OUTPTK                                                    OUTPTK          1     A
      OUTPTK              IDENT  OUTPTK                                            OUTPTK          2     A
      OUTPTK    B=OUTPK  RPVDEF                                                    OUTPTK          3           I
               -CCG27
      OUTPTK    OUTPTK   RPVDEF                                                    CCG27           5     A
      OUTPTK              SST                                                      OUTPTK          4     A
      OUTPTK              LIST   F                                                 OUTPTK          5     A
      OUTPTK                                                                       OUTPTK          6     A
      OUTPTK    OUTPTK   TITLE  OUTPTK -    COMBINED 'KODER'-'OUTPTC' FACILITY     OUTPTK          7     A
      OUTPTK                                                                       OUTPTK          8     A
      OUTPTK    .TEST    IFNE   TEST,0                                             OUTPTK          9     A
      OUTPTK              MACRO  FORM,NAME                                         OUTPTK         10     A
      OUTPTK   NAME       BSS    0                                                 OUTPTK         11     A
      OUTPTK   NAME()     SET    *                                                 OUTPTK         12     A
      OUTPTK    FORM      ENDM                                                     OUTPTK         13     A
      OUTPTK                                                                       OUTPTK         14     A
      OUTPTK   INVOKE     MACRO  FORM                                              OUTPTK         15     A
      OUTPTK              SB7    FORM                                              OUTPTK         16     A
      OUTPTK              RJ     RETURN                                            OUTPTK         17     A
      OUTPTK              RMT                                                      OUTPTK         18     A
      OUTPTK              IF     -DEF,FORM(),1                                     OUTPTK         19     A
      OUTPTK              ERR    FORM NOT INVOKABLE, CHECK CROSS REFERENCES AND CHANGE OUTPTK     20     A
      OUTPTK              RMT                                                      OUTPTK         21     A

               0         1         2         3         4         5         6         7         8
               12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OUTPTK

```
OUTPTK   INVOKE    ENDM                                                    OUTPTK   22   A
OUTPTK    LOCN     SPACE  3                                                OUTPTK   23   A
OUTPTK   LOCN      MACRO  CHAR,ADDRESS                                     OUTPTK   24   A
OUTPTK   CHARNOW   MICRO  1,0, CHAR                                        OUTPTK   25   A
OUTPTK   LASTCHAR  MICRO  2,1, CHAR                                        OUTPTK   26   A
OUTPTK             IFC    EQ, "LASTCHAR"  ,1                              OUTPTK   27   A
OUTPTK   CHARNOW   MICRO  1,0, 1R_CHAR                                     OUTPTK   28   A
OUTPTK   *                                                                 OUTPTK   29   A
OUTPTK    .A       IFLE   "CHARNOW",LASTCHAR                               OUTPTK   30   A
OUTPTK             ERR    (CHAR) - FORMAT CHARACTER OUT OF SEQUENCE "SEQUENCE".  OUTPTK   31   A
OUTPTK    .A       ELSE                                                    OUTPTK   32   A
OUTPTK    .B       DUP    "CHARNOW"-LASTCHAR-1                            OUTPTK   33   A
OUTPTK             IFEQ   $,59,2                                          OUTPTK   34   A
OUTPTK             JP     ILLEGAL                                         OUTPTK   35   A
OUTPTK             SKIP   1                                               OUTPTK   36   A
OUTPTK   -         JP     ILLEGAL                                         OUTPTK   37   A
OUTPTK    .B       ENDD                                                    OUTPTK   38   A
OUTPTK    .A       ENDIF                                                   OUTPTK   39   A
OUTPTK   *                                                                 OUTPTK   40   A
OUTPTK             IFEQ   $,59,2                                          OUTPTK   41   A
OUTPTK             JP     ADDRESS                                         OUTPTK   42   A
OUTPTK             SKIP   1                                               OUTPTK   43   A
OUTPTK   -         JP     ADDRESS                                         OUTPTK   44   A
OUTPTK   *                                                                 OUTPTK   45   A
OUTPTK    LASTCHAR SET    "CHARNOW"                                        OUTPTK   46   A
OUTPTK    LOCN     ENDM                                                    OUTPTK   47   A
OUTPTK                                                                     OUTPTK   48   A
OUTPTK                                                                     OUTPTK   49   A
OUTPTK   LASTCHAR  SET    -1                                               OUTPTK   50   A
OUTPTK    BSS      OPSYN  BSSZ                                             OUTPTK   51   A
OUTPTK             TITLE                  SYMBOL AND STORAGE DEFINITION    OUTPTK   52   A
OUTPTK   DIGMASK   DATA   00000000077740000000B                            OUTPTK   53   A
OUTPTK    NEEDW    BSS    0                                               OUTPTK   54   A
OUTPTK             ECHO   2,L=(A,I,L,O,R,T,Z)                             OUTPTK   55   A
OUTPTK             POS    60-1R_L                                         OUTPTK   56   A
OUTPTK             VFD    1/1                                             OUTPTK   57   A
OUTPTK             VFD    *P/0                                            OUTPTK   58   A
OUTPTK   TEMPA0.   BSS    1                                               OUTPTK   59   A
OUTPTK   APLIST    BSS    1                                               OUTPTK   60   A
OUTPTK   R         BSS    1               FIELD DESCRIPTOR REPEAT COUNT   OUTPTK   61   A
OUTPTK   W         BSS    1               WIDTH                           OUTPTK   62   A
OUTPTK   GOCELL    BSS    1                                               OUTPTK   63   A
OUTPTK   GRC       BSS    1               GROUP REPEAT COUNT FOR CURRENT GROUP  OUTPTK   64   A
OUTPTK   FORMAT    BSS    1                                               OUTPTK   65   A
OUTPTK   LEVEL     BSS    1                                               OUTPTK   66   A
OUTPTK   GROUPADR  BSS    1                                               OUTPTK   67   A
OUTPTK   HCOUNT    BSS    1                                               OUTPTK   68   A
OUTPTK   MAXLEVEL  EQU    2               THE MAXIMUM NUMBER OF GROUP LEVELS  OUTPTK   69   A
OUTPTK   LEVSTACK  BSS    MAXLEVEL+1      STORAGE FOR GROUP LEVEL STACK   OUTPTK   70   A
OUTPTK   UNIT      BSS    1               HOLDS I/O UNIT DESIGNATION      OUTPTK   71   A
OUTPTK   MAXCOUNT  BSS    1                                               OUTPTK   72   A
OUTPTK    RESCAN   BSS    1               HOLDS LEVEL STACK INFORMATION FOR  OUTPTK   73   A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OUTPTK

```
OUTPTK                                  RESCANNING THE FORMAT WHEN THE                OUTPTK        74     A
OUTPTK                                  TERMINAL RIGHT PARENTHESIS IS REACHED         OUTPTK        75     A
OUTPTK   PARENFLG  BSS    1                                                           OUTPTK        76     A
OUTPTK   RESETLIN  BSS    1                                                           OUTPTK        77     A
OUTPTK   RESETFMT  BSS    1                                                           OUTPTK        78     A
OUTPTK   RESETX3   BSS    1               TEMPORARY STORAGE USED BY 'SLASH'           OUTPTK        79     A
OUTPTK   RESETITM  BSS    1               TEMPORARY STORAGE USED BY 'SLASH'           OUTPTK        80     A
OUTPTK                                                                                OUTPTK        81     A
OUTPTK   OUTADDR   VFD    60/BUFFER       BEGINNING OF OTUPUT LINE                    OUTPTK        82     A
OUTPTK             USE    /LINE/                                                      OUTPTK        83     A
OUTPTK    CONTINUE BSSZ   1          * EXPLICIT LINE TERMINATION MODE FLAG *          OUTPTK        84     A
OUTPTK             VFD    12/2000B,48/BUFFER        VALUE TO SET TO FOR CONT MODE     OUTPTK        85     A
OUTPTK   BUFFER    BSSZ   16                                                          OUTPTK        86     A
OUTPTK             USE    *                                                           OUTPTK        87     A
OUTPTK                                                                                OUTPTK        88     A
OUTPTK   FORMS     BSS    0                                                           OUTPTK        89     A
OUTPTK             LOCN   A,AFORMP        ALPHANUMERIC (LEFT JUSTIFIED)               OUTPTK        90     A
OUTPTK             LOCN   H,HFORM         HOLLERITH STRING PROCESSING (COUNT)         OUTPTK        91     A
OUTPTK             LOCN   I,IFORM         INTEGER CONVERSION                          OUTPTK        92     A
OUTPTK             LOCN   L,LFORM                                                     OUTPTK        93     A
OUTPTK             LOCN   O,OFORM         OCTAL CONVERSION                            OUTPTK        94     A
OUTPTK             LOCN   R,RFORM         ALPHANUMERIC (RIGHT JUSTIFIED)              OUTPTK        95     A
OUTPTK             LOCN   T,TFORM         TABULATE (TEXT EDITING)                     OUTPTK        96     A
OUTPTK             LOCN   X,XFORM         SPACING  (TEXT EDITING)                     OUTPTK        97     A
OUTPTK             LOCN   Z,ZFORM     OCTAL CONVERSION ( LEADING BLANK FILL )         OUTPTK        98     A
OUTPTK             LOCN   *,DELIM         HOLERITH STRING (DELIMITED)                 OUTPTK        99     A
OUTPTK             LOCN   /,SLASH         TERMINATE CURRENT RECORD                    OUTPTK       100     A
OUTPTK             LOCN   1R(,LPFORM      BEGIN FIELD DESCRIPTOR GROUPING             OUTPTK       101     A
OUTPTK             LOCN   1R),RPFORM      END FIELD DESCRIPTOR GROUPING               OUTPTK       102     A
OUTPTK             LOCN   55B,RETURNX                                                 OUTPTK       103     A
OUTPTK             LOCN   (,),NEXTDESC    FIELD DESCRIPTOR SEPARATOR                  OUTPTK       104     A
OUTPTK             LOCN   64B,DELIM       HOLLERITH STRING (DELIMITED)                OUTPTK       105     A
OUTPTK             TITLE                  INITIALIZATION INTERFACE                    OUTPTK       106     A
OUTPTK   ***********************************************************************    OUTPTK       107     A
OUTPTK   *                                                                     *    OUTPTK       108     A
OUTPTK   *                                                                     *    OUTPTK       109     A
OUTPTK   *                                                                     *    OUTPTK       110     A
OUTPTK   *               O U T C I . -- INITIAL ENTRY POINT                    *    OUTPTK       111     A
OUTPTK   *                                                                     *    OUTPTK       112     A
OUTPTK   *                                                                     *    OUTPTK       113     A
OUTPTK   *               SAVE THE FORMAT ADDRESS, THE UNIT NUMBER AND SETUP    *    OUTPTK       114     A
OUTPTK   *             FOR PROCESSING THE BEGINNING OF AN I/O SEQUENCE.        *    OUTPTK       115     A
OUTPTK   *                                                                     *    OUTPTK       116     A
OUTPTK   *                                                                     *    OUTPTK       117     A
OUTPTK   ***********************************************************************    OUTPTK       118     A
OUTPTK             SPACE  1                                                           OUTPTK       119     A
OUTPTK   OUTCI.    ENTRY.                                                             OUTPTK       120     A
OUTPTK             SB1    1                                                           OUTPTK       121     A
OUTPTK             SX6    A0                                                          OUTPTK       122     A
OUTPTK             SA2    A1+B1           GET APLIST FORMAT WORD.                     OUTPTK       123     A
OUTPTK             SA6    TEMPA0.         SAVE FOR LATER RESTORATION.                 OUTPTK       124     A
OUTPTK             SX7    B1                                                          OUTPTK       125     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OUTPTK

```
OUTPTK          SX6    X2                                                  OUTPTK     126    A
OUTPTK          BX4    X6                                                  OUTPTK     127    A
OUTPTK          SA7    GRC          SET THE GROUP REPEAT COUNT 'GRC' TO ONE OUTPTK     128    A
OUTPTK                                FOR THE OUTER PARENTHESIS LEVEL.    OUTPTK     129    A
OUTPTK          SPACE  1                                                   OUTPTK     130    A
OUTPTK          SA6    FORMAT       SAVE THE FORMAT PARAMETER ADDRESS IN  OUTPTK     131    A
OUTPTK                                'FORMAT'.                           OUTPTK     132    A
OUTPTK          SPACE  1                                                   OUTPTK     133    A
OUTPTK          LX2    2            DETERMINE THE ADDRESS OF THE FIRST WORD OUTPTK     134    A
OUTPTK          NG     X2,INIT.1      OF THE FORMAT FROM THE 'VAR' BIT IN OUTPTK     135    A
OUTPTK          IX4    X6+X7          THE APLIST ENTRY.  IF THE BIT IS NOT OUTPTK     136    A
OUTPTK                                SET, THE ADDRESS IS INCREMENTED BY 1 OUTPTK     137    A
OUTPTK                                TO YIELD THE TRUE FORMAT POINTER.   OUTPTK     138    A
OUTPTK          SPACE  1                                                   OUTPTK     139    A
OUTPTK  INIT.1  SX3    X1           DETERMINE THE UNIT NUMBER OF THE FILE OUTPTK     140    A
OUTPTK          PL     X1,INIT.2      TO BE ACCESSED.  IF THE 'VAR' BIT IS OUTPTK     141    A
OUTPTK          SA3    X1             SET, THE ADDRESS POINTS TO A WORD   OUTPTK     142    A
OUTPTK                                CONTAINING THE UNIT NUMBER.         OUTPTK     143    A
OUTPTK                                OTHERWISE THE UNIT NUMBER IS        OUTPTK     144    A
OUTPTK                                CONTAINED IN THE APLIST WORD ENTRY. OUTPTK     145    A
OUTPTK          SPACE  1                                                   OUTPTK     146    A
OUTPTK  INIT.2  SX6    A1+2                                               OUTPTK     147    A
OUTPTK          SA6    APLIST       SAVE THE APLIST POINTER FOR LATER     OUTPTK     148    A
OUTPTK                                PROCESSING OF THE LIST ITEMS.       OUTPTK     149    A
OUTPTK          SPACE  1                                                   OUTPTK     150    A
OUTPTK          BX6    X3                                                  OUTPTK     151    A
OUTPTK          SX5    60-2*6       SET THE INITIAL SHIFT COUNT FOR THE   OUTPTK     152    A
OUTPTK          SB2    X5             FORMAT TO POINT JUST BEYOND THE     OUTPTK     153    A
OUTPTK                                INITIAL LEFT PARENTHESIS.           OUTPTK     154    A
OUTPTK          SA6    UNIT           (SAVE THE UNIT DESIGNATION.)        OUTPTK     155    A
OUTPTK          LX5    18                                                  OUTPTK     156    A
OUTPTK          IX6    X5+X4                                               OUTPTK     157    A
OUTPTK          SA2    X4           FETCH THE FIRST FORMAT WORD.          OUTPTK     158    A
OUTPTK          SB6    60                                                  OUTPTK     159    A
OUTPTK          SA6    GROUPADR     SAVE THE POINTERS TO THE OUTER        OUTPTK     160    A
OUTPTK                                PARENTHESIS GROUP IN 'GROUPADR'.    OUTPTK     161    A
OUTPTK          SA5    OUTADDR                                             OUTPTK     162    A
OUTPTK          LX6    18                                                  OUTPTK     163    A
OUTPTK          IX7    X6+X7                                               OUTPTK     164    A
OUTPTK          SA7    LEVSTACK     SETUP THE INITIAL LEVEL STACK ENTRY-- OUTPTK     165    A
OUTPTK          SA7    RESCAN         THE LEVEL STACK CONTAINS THE        OUTPTK     166    A
OUTPTK                                NECESSARY INFORMATION FOR RESTARTING OUTPTK     167    A
OUTPTK                                THE FORMAT SCAN AT THE BEGINNING OF OUTPTK     168    A
OUTPTK                                A PARENTHESIS GROUP, THE ZERO LEVEL OUTPTK     169    A
OUTPTK                                ENTRY MUST THEREFORE CONTAIN THE    OUTPTK     170    A
OUTPTK                                INFORMATION FOR RESTARTING THE FORM OUTPTK     171    A
OUTPTK                                SCAN WHEN THE FINAL RIGHT PARENTHESIS OUTPTK     172    A
OUTPTK                                IN THE FORMAT IS ENCOUNTERED.  THE  OUTPTK     173    A
OUTPTK                                INFORMATION FOR THE ZERO ENTRY IS   OUTPTK     174    A
OUTPTK                                     SHIFTCT = 48, POSITION AFTER '(' OUTPTK     175    A
OUTPTK                                     WORD    = ADDRESS OF FIRST     OUTPTK     176    A
OUTPTK                                               FORMAT WORD          OUTPTK     177    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OUTPTK

```
OUTPTK                                        GRC    = 1, REMAINING REPEAT          OUTPTK     178    A
OUTPTK                                                 COUNT FOR GROUP              OUTPTK     179    A
OUTPTK            SA3    CONTINUE                                                   OUTPTK     180    A
OUTPTK            PX5    X5                                                         OUTPTK     181    A
OUTPTK            BX6    X6-X6                                                      OUTPTK     182    A
OUTPTK            SA6    LEVEL          INITIALIZE THE LEVEL NUMBER FOR THE         OUTPTK     183    A
OUTPTK                                     LEVEL STACK.                             OUTPTK     184    A
OUTPTK            SPACE  1                                                          OUTPTK     185    A
OUTPTK            SB4    B0             SET THE ITEM COUNT TO ZERO                  OUTPTK     186    A
OUTPTK            SA6    PARENFLG       TURN OF THE TERMINAL RIGHT PAREN FLAG       OUTPTK     187    A
OUTPTK            SPACE  1                                                          OUTPTK     188    A
OUTPTK            ZR     X3,INIT.3      IF THE PROCESSING MODE IS 'EXPLICIT         OUTPTK     189    A
OUTPTK            SA4    MAXCOUNT         LINE TERMINATION' THEN CONTINUE THE       OUTPTK     190    A
OUTPTK            LX5    X3               LAST LINE WHERE THE PRECEDING WRITE       OUTPTK     191    A
OUTPTK            BX6    X4               STATEMENT FINISHED (I.E. IF               OUTPTK     192    A
OUTPTK                                    'CONTINUE' IS NON-ZERO THEN IT           OUTPTK     193    A
OUTPTK                                    CONTAINS THE LINE INITIALIZING           OUTPTK     194    A
OUTPTK                                    INFORMATION, 'MAXCOUNT' SHOULD REMAIN    OUTPTK     195    A
OUTPTK                                    UNCHANGED IF 'CONTINUE' IS NON-ZERO).    OUTPTK     196    A
OUTPTK            SPACE  1                                                          OUTPTK     197    A
OUTPTK   INIT.3   SA6    MAXCOUNT       SET THE MAXIMUM LENGTH ATTAINED FOR THE     OUTPTK     198    A
OUTPTK                                     LINE (IF A TAB BACKWARD HAS OCCURRED)    OUTPTK     199    A
OUTPTK            SPACE  1                                                          OUTPTK     200    A
OUTPTK            UX0    X5,B3          SET THE LINE POSITION POINTERS AND THEN     OUTPTK     201    A
OUTPTK            ZR     NEXTDESC         GO TO 'NEXTDESC' TO BEGIN PORCESSING.     OUTPTK     202    A
OUTPTK            EJECT                                                             OUTPTK     203    A
OUTPTK   **       OUTCR. - ITEM CONVERSION                                          OUTPTK     204    A
OUTPTK   *                                                                  *       OUTPTK     205    A
OUTPTK   *        REGISTER CONVENTIONS --                                   *       OUTPTK     206    A
OUTPTK   *                                                                  *       OUTPTK     207    A
OUTPTK   *        B0 -- 0            A0 -- UNUSED        X0 -- OUTPUT WORD  *       OUTPTK     208    A
OUTPTK   *        B1 -- 1            A1 -- DATA ADDRESS  X1 -- DATA ITEM    *       OUTPTK     209    A
OUTPTK   *        B2 -- FORMAT SHIFT A2 -- FORMAT ADDR   X2 -- FORMAT WORD  *       OUTPTK     210    A
OUTPTK   *        B3 -- BITS FILLED  A3 -- SCRATCH       X3 -- SPECIAL HOLD  *       OUTPTK     211    A
OUTPTK   *        B4 -- ITEM COUNT   A4 -- SCRATCH       X4 -- SCRATCH       *       OUTPTK     212    A
OUTPTK   *        B5 -- SCRATCH      A5 -- SCRATCH       X5 -- SCRATCH       *       OUTPTK     213    A
OUTPTK   *        B6 -- 60           A6 -- SCRATCH       X6 -- SCRATCH       *       OUTPTK     214    A
OUTPTK   *        B7 -- SCRATCH      A7 -- SCRATCH       X7 -- SCRATCH       *       OUTPTK     215    A
OUTPTK            SPACE  1                                                          OUTPTK     216    A
OUTPTK   OUTPTCX  PX6    X0,B3          SAVE THE CURRENT POINTERS TO THE OUTPUT     OUTPTK     217    A
OUTPTK            SX5    A2               LINE AND THE FORMAT.                      OUTPTK     218    A
OUTPTK            SA6    RESETLIN                                                   OUTPTK     219    A
OUTPTK            PX7    X5,B2                                                      OUTPTK     220    A
OUTPTK            SB5    B1                                                         OUTPTK     221    A
OUTPTK            SA7    RESETFMT                                                   OUTPTK     222    A
OUTPTK            SA2    TEMPA0.                                                    OUTPTK     223    A
OUTPTK            SB2    B1                                                         OUTPTK     224    A
OUTPTK            SA0    X2                                                         OUTPTK     225    A
OUTPTK            SPACE  1                                                          OUTPTK     226    A
OUTPTK            SA2    APLIST                                                     OUTPTK     227    A
OUTPTK            SA3    X2             OBTAIN APLIST ITEM WORD ENTRY.              OUTPTK     228    A
OUTPTK            SPACE  1                                                          OUTPTK     229    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OUTPTK

```
OUTPTK            SB1    -B1            PRESET IN CASE OF TERMINAL CALL.          OUTPTK   230   A
OUTPTK            NZ     X3,PRESET      IF NOT END OF I/O PARAMETER LIST.         OUTPTK   231   A
OUTPTK            PL     X3,PRESET.2      ELSE GO TO 'PRESET.2' TO HANDLE THE     OUTPTK   232   A
OUTPTK                                     END OF I/O LIST PROCESSING.           OUTPTK   233   A
OUTPTK            EQ     OUTCI.                                                   OUTPTK   234   A
OUTPTK                                                                            OUTPTK   235   A
OUTPTK    OUTCR.  ENTRY.                   ENTRY POINT FOR INTERRUPTED I/O LIST  OUTPTK   236   A
OUTPTK            SA2    APLIST         MAKE THINGS LOOK AS IF THERE WAS NOT AN   OUTPTK   237   A
OUTPTK            SA5    OUTCR.           INTERRUPTION.                           OUTPTK   238   A
OUTPTK            LX7    X5                                                       OUTPTK   239   A
OUTPTK            SX2    A1                                                       OUTPTK   240   A
OUTPTK            SA7    OUTCI.                                                   OUTPTK   241   A
OUTPTK            BX3    X1                                                       OUTPTK   242   A
OUTPTK            SB1    -1                                                       OUTPTK   243   A
OUTPTK                                                                            OUTPTK   244   A
OUTPTK                                                                            OUTPTK   245   A
OUTPTK    PRESET  ZR     X3,PRESET.1      IF TERMINAL CALL                        OUTPTK   246   A
OUTPTK            IX4    X3+X3                                                    OUTPTK   247   A
OUTPTK            SB1    X3                 (B1) = DATA ITEM ADDRESS              OUTPTK   248   A
OUTPTK            PL     X4,PRESET.1      IF ITEM IS ^ F.P.                       OUTPTK   249   A
OUTPTK            SPACE  1                                                        OUTPTK   250   A
OUTPTK            AX4    1                                                        OUTPTK   251   A
OUTPTK            MX0    60-6                                                     OUTPTK   252   A
OUTPTK            BX6    -X0*X4                                                   OUTPTK   253   A
OUTPTK            LX4    60-6                                                     OUTPTK   254   A
OUTPTK            SB3    X6                                                       OUTPTK   255   A
OUTPTK            SA5    A0+B3          COMPUTE THE DATA ITEM ADDRESS BY ADDING   OUTPTK   256   A
OUTPTK            SB6    X4               THE FP OFFSET TO THE APPROPRIATE FP     OUTPTK   257   A
OUTPTK            SB1    X5+B6            ADDRESS IN THE SUBROUTINE APLIST.       OUTPTK   258   A
OUTPTK            SPACE  1                                                        OUTPTK   259   A
OUTPTK    PRESET.1 SX7     X2+1           INCREMENT THE APLIST POINTER, AND SET   OUTPTK   260   A
OUTPTK            AX3    24               THE ITEM COUNT INTO B2.  A ZERO ITEM    OUTPTK   261   A
OUTPTK            SA7    A2               COUNT WILL BE TREATED AS AN ITEM        OUTPTK   262   A
OUTPTK            SB2    X3               COUNT OF 1.                             OUTPTK   263   A
OUTPTK            NZ     B2,PRESET.2                                              OUTPTK   264   A
OUTPTK            SB2    1                                                        OUTPTK   265   A
OUTPTK            SPACE  1                                                        OUTPTK   266   A
OUTPTK    PRESET.2 SA5     RESETLIN                                               OUTPTK   267   A
OUTPTK            SA4    RESETFMT                                                 OUTPTK   268   A
OUTPTK            PL     B1,SETUP.1     IF THIS IS NOT A FINAL ENTRY THEN GO      OUTPTK   269   A
OUTPTK            SB2    B1               TO 'SETUP.1', OTHERWISE SWITCH THE      OUTPTK   270   A
OUTPTK            SB1    -B1              FINAL INDICATOR FOR A LATER CHECK.      OUTPTK   271   A
OUTPTK            SPACE  1                                                        OUTPTK   272   A
OUTPTK    SETUP.1 SA1    B1-1           SET THE BASE DATA ITEM ADDRESS (I.E.      OUTPTK   273   A
OUTPTK                                     POINT TO THE WORD IMMEDIATELY PRIOR   OUTPTK   274   A
OUTPTK                                     TO THE SET OF DATA ITEMS).            OUTPTK   275   A
OUTPTK            SPACE  1                                                        OUTPTK   276   A
OUTPTK            UX0    X5,B3          SETUP THE POINTERS TO THE LINE.           OUTPTK   277   A
OUTPTK            SA2    X4             PICK UP THE CURRENT FORMAT WORD.          OUTPTK   278   A
OUTPTK            SB4    B2             SETUP THE COUNT OF THE NUMBER OF ITEMS    OUTPTK   279   A
OUTPTK                                     TO CONVERT.                           OUTPTK   280   A
OUTPTK            SB1    1                                                        OUTPTK   281   A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OUTPTK

```
OUTPTK            UX4    X4,B2           SETUP THE FORMAT SHIFT COUNT POINTER9          OUTPTK    282    A
OUTPTK            SB6    60                                                             OUTPTK    283    A
OUTPTK            SA5    PARENFLG                                                       OUTPTK    284    A
OUTPTK            PL     B4,INTER        IF THIS IS AN INTERMEDIATE ENTRY THEN          OUTPTK    285    A
OUTPTK                                     GO TO !INTER' TO ENTER THE                   OUTPTK    286    A
OUTPTK                                     PROCESSING CYCLE.                            OUTPTK    287    A
OUTPTK            SA5    CONTINUE                                                       OUTPTK    288    A
OUTPTK            SA4    OUTADDR                                                        OUTPTK    289    A
OUTPTK            NZ     X5,SETUP.2      IF THE PROCESSING MODE IS 'EXPLICIT            OUTPTK    290    A
OUTPTK                                     LINE TERMINATION' THEN GO TO                 OUTPTK    291    A
OUTPTK                                     'SETUP.2'.                                   OUTPTK    292    A
OUTPTK            SPACE  1                                                              OUTPTK    293    A
OUTPTK            SA5    MAXCOUNT        OTHERWISE INVOKE 'SLASH' TO DUMP THE           OUTPTK    294    A
OUTPTK            IX3    X0-X4             CURRENT OUTPUT LINE IF IT IS NOT             OUTPTK    295    A
OUTPTK            SX6    X3+B3             EMPTY.                                       OUTPTK    296    A
OUTPTK            BX7    X6+X5                                                          OUTPTK    297    A
OUTPTK            ZR     X7,OUTCI.       WHEN (IF NECESSARY) THE OUTPUT LINE HAS        OUTPTK    298    A
OUTPTK            INVOKE SLASH           BEEN DUMPED, RETURN TO THE CALLER.             OUTPTK    299    A
OUTPTK            EQ     OUTCI.                                                         OUTPTK    300    A
OUTPTK            SPACE  1                                                              OUTPTK    301    A
OUTPTK   SETUP.2  PX6    X0,B3           SINCE 'EXPLICIT LINE TERMINATION' IS IN        OUTPTK    302    A
OUTPTK            NO                       FORCE, SAVE THE CURRENT LINE POINTERS        OUTPTK    303    A
OUTPTK            SA6    CONTINUE          AND  RETURN TO THE CALLER.                   OUTPTK    304    A
OUTPTK            EQ     OUTCI.                                                         OUTPTK    305    A
OUTPTK            SPACE  1                                                              OUTPTK    306    A
OUTPTK   INTER    BX7    X7-X7                                                          OUTPTK    307    A
OUTPTK            SA7    A5              CLEAR THE PAREN FLAG.                          OUTPTK    308    A
OUTPTK            ZR     X5,IFNOMORE     GO TO 'IFNOMORE' IF THE LAST ENTRY WAS         OUTPTK    309    A
OUTPTK                                     NOT STOPPED BY THE TERMINAL PAREN,           OUTPTK    310    A
OUTPTK            EQ     NEXTDESC                                                       OUTPTK    311    A
OUTPTK            TITLE                  A -, R - FORMAT PROCESSING SECTIONS             OUTPTK    312    A
OUTPTK   ************************************************************************      OUTPTK    313    A
OUTPTK   *                                                                      *      OUTPTK    314    A
OUTPTK   *                      R - FORMAT PROCESSING                           *      OUTPTK    315    A
OUTPTK   *                                                                      *      OUTPTK    316    A
OUTPTK   *              CONVERT THE R-FORM DATA ITEM INTO A-FORM AND THEN USE *        OUTPTK    317    A
OUTPTK   *            THE A-FORMAT PROCESSING SECTION TO PROCESS THE ITEM.     *      OUTPTK    318    A
OUTPTK   *                                                                      *      OUTPTK    319    A
OUTPTK   ************************************************************************      OUTPTK    320    A
OUTPTK            SPACE  1                                                              OUTPTK    321    A
OUTPTK   RFORM    FORM                                                                  OUTPTK    322    A
OUTPTK            SB5    X6                IF THE FIELD WIDTH IS LESS THE ELEVEN        OUTPTK    323    A
OUTPTK            LE     B5,B6,RFORM.1       CHARACTERS, THEN GO TO RFORM.1,            OUTPTK    324    A
OUTPTK            SX6    B5-B6             OTHERWISE CALL SPACE TO SPACE                OUTPTK    325    A
OUTPTK            CALL   SPACE               THE EXCESS NUMBER OF POSITIONS             OUTPTK    326    A
OUTPTK            SX6    B6                                                             OUTPTK    327    A
OUTPTK   RFORM.1  SB7    X6                CALCULATE THE SHIFT COUNT NECESSARY T        OUTPTK    328    A
OUTPTK            SB5    B6-B7             TRANSFORM THE DATA ITEM INTO A-FORM          OUTPTK    329    A
OUTPTK            LX1    X1,B5             AND THEN PERFORM THE TRANSFORMATION.         OUTPTK    330    A
OUTPTK            EQ     AFORM                                                          OUTPTK    331    A
OUTPTK                                                                                 OUTPTK    332    A
OUTPTK   ************************************************************************      OUTPTK    333    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OUTPTK

```
OUTPTK   *                                                                *          OUTPTK        334     A
OUTPTK   *                      A - FORMAT PROCESSING                     *          OUTPTK        335     A
OUTPTK   *                                                                *          OUTPTK        336     A
OUTPTK   *                 IF THE NUMBER OF CHARACTERS TO BE TRANSMITTED IS  *          OUTPTK        337     A
OUTPTK   *              GREATER THAN TEN, CALL SPACE TO INSERT THE CORRECT  *          OUTPTK        338     A
OUTPTK   *              NUMBER OF BLANKS THEN GO TO AFORM.                  *          OUTPTK        339     A
OUTPTK   *                 THIS A FORMAT PROCESSOR IS CALLED ONLY WHEN THE  *          OUTPTK        340     A
OUTPTK   *              A-FORMAT ACTUALLY APPEARS IN THE USER FORMAT.  THE OTHER *     OUTPTK        341     A
OUTPTK   *              A-FORMAT PROCESSOR IS ONLY USED INTERNALLY.         *          OUTPTK        342     A
OUTPTK                                                                                 OUTPTK        343     A
OUTPTK    AFORMP   FORM                                                                OUTPTK        344     A
OUTPTK             SB5    X6                IF THE FIELD WIDTH IS LESS THE ELEVEN       OUTPTK        345     A
OUTPTK             LE     B5,B6,AFORM         CHARACTERS, THEN GO TO AFORM,             OUTPTK        346     A
OUTPTK             SX6    B5-B6             OTHERWISE CALL SPACE TO SPACE               OUTPTK        347     A
OUTPTK             CALL   SPACE               THE EXCESS NUMBER OF POSITIONS            OUTPTK        348     A
OUTPTK             SX6    B6                                                          OUTPTK        349     A
OUTPTK                                                                                 OUTPTK        350     A
OUTPTK   ************************************************************************     OUTPTK        351     A
OUTPTK   *                                                                *          OUTPTK        352     A
OUTPTK   *                      A - FORMAT PROCESSING                     *          OUTPTK        353     A
OUTPTK   *                                                                *          OUTPTK        354     A
OUTPTK   *                 IF THE NUMBER OF CHARACTERS TO BE TRANSMITTED FROM  *        OUTPTK        355     A
OUTPTK   *              THE A-FORM DATA WILL FIT INTO THE SPACE REMAINING IN THE *    OUTPTK        356     A
OUTPTK   *              CURRENT OUTPUT WORD, THEN THE SPACE IN THE OUTPUT WORD  *     OUTPTK        357     A
OUTPTK   *              NECESSARY TO ACCOMODATE THE A-FORM DATA IS CLEARED AND  *     OUTPTK        358     A
OUTPTK   *              THE A-FORM DATA INSERTED.  IF NOT ENOUGH SPACE IS       *     OUTPTK        359     A
OUTPTK   *              AVAILABLE IN THE CURRENT OUTPUT WORD FOR THE A-FORM     *     OUTPTK        360     A
OUTPTK   *              DATA, THEN THE DATA IS SPLIT TO OCCUPY THE END OF THE   *     OUTPTK        361     A
OUTPTK   *              CURRENT OUTPUT WORD AND THE BEGINNING OF THE NEXT       *     OUTPTK        362     A
OUTPTK   *              OUTPUT WORD.                                            *     OUTPTK        363     A
OUTPTK   *                                                                *          OUTPTK        364     A
OUTPTK   ************************************************************************     OUTPTK        365     A
OUTPTK             SPACE  1                                                            OUTPTK        366     A
OUTPTK   AFORM     FORM                                                                OUTPTK        367     A
OUTPTK             SB5    B6-B3           IF THE A-FORM DATA MUST BE SPLIT TO         OUTPTK        368     A
OUTPTK             SB7    X6                OCCUPY TWO OUTPUT WORDS THEN GO           OUTPTK        369     A
OUTPTK             LT     B5,B7,AFORM2WD    TO 'AFORM2WD'.                            OUTPTK        370     A
OUTPTK             SPACE  1                                                            OUTPTK        371     A
OUTPTK   AFORM.1   MX4    1               FORM THE MASK NECESSARY TO EXTRACT          OUTPTK        372     A
OUTPTK             SB7    B7-B1             THE A-FORM DATA AND INSERT IT INTO        OUTPTK        373     A
OUTPTK             SA5    X0                THE OUTPUT WORD.                          OUTPTK        374     A
OUTPTK             AX7    X4,B7                                                       OUTPTK        375     A
OUTPTK             BX6    X7*X1             (EXTRACT THE A-FORM DATA)                 OUTPTK        376     A
OUTPTK             LX5    X5,B3                                                       OUTPTK        377     A
OUTPTK             SB7    B3+B7                                                       OUTPTK        378     A
OUTPTK             BX4    -X7*X5            (CLEAR SPACE IN THE OUTPUT WORD)          OUTPTK        379     A
OUTPTK             SB3    B7+B1             (INCREMENT THE COUNT OF BITS IN THE       OUTPTK        380     A
OUTPTK                                      CURRENT OUTPUT WORD)                      OUTPTK        381     A
OUTPTK             SPACE  1                                                            OUTPTK        382     A
OUTPTK             BX6    X4+X6             (INSERT THE A-FORM DATA, THE BOOLEAN      OUTPTK        383     A
OUTPTK                                      UNIT MUST BE USED IN CASE AN ENTIRE       OUTPTK        384     A
OUTPTK                                      OUTPUT WORD CONSISTS OF SEMI-COLONS--     OUTPTK        385     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OUTPTK

```
OUTPTK                                  SINCE THE OUTPUT WORD WOULD BECOME          OUTPTK     386    A
OUTPTK                                  POSITIVE ZERO--NOT MINUS ZERO--IF           OUTPTK     387    A
OUTPTK                                  THE LONG ADD UNIT IS USED.)                 OUTPTK     388    A
OUTPTK            LX7    X6,B5                                                      OUTPTK     389    A
OUTPTK            SA7    A5                                                         OUTPTK     390    A
OUTPTK            ZR     RETURNX        GO TO 'RETURNX'.                            OUTPTK     391    A
OUTPTK            SPACE  2                                                          OUTPTK     392    A
OUTPTK   AFORM2WD MX4    1              EXTRACT THE NUMBER OF CHARACTERS IN         OUTPTK     393    A
OUTPTK            SB3    B3-B1            A-FORM NECESSARY TO FILL THIS OUTPUT      OUTPTK     394    A
OUTPTK            LX1    X1,B5            WORD.                                     OUTPTK     395    A
OUTPTK            SB7    B7-B5                                                      OUTPTK     396    A
OUTPTK            SB5    B6                                                         OUTPTK     397    A
OUTPTK            SA5    X0                                                         OUTPTK     398    A
OUTPTK            AX7    X4,B3                                                      OUTPTK     399    A
OUTPTK            BX6    -X7*X1                                                     OUTPTK     400    A
OUTPTK            SB3    B0                                                         OUTPTK     401    A
OUTPTK            SX0    X0+B1                                                      OUTPTK     402    A
OUTPTK            BX5    X7*X5          CLEAR SPACE IN THE CURRENT OUTPUT WORD      OUTPTK     403    A
OUTPTK                                  TO ACCOMODATE THE AFORM DATA.               OUTPTK     404    A
OUTPTK            SPACE  1                                                          OUTPTK     405    A
OUTPTK            BX6    X5+X6                                                      OUTPTK     406    A
OUTPTK            SA6    A5             APPEND THE AFORM DATA TO THE CURRENT        OUTPTK     407    A
OUTPTK            ZR     AFORM.1          OUTPUT WORD AND THEN GO TO 'AFORM.1'      OUTPTK     408    A
OUTPTK                                  TO PROCESS THE REMAINING CHARACTERS.        OUTPTK     409    A
OUTPTK            TITLE                  I - FORMAT PROCESSING SECTION               OUTPTK     410    A
OUTPTK   ***********************************************************************   OUTPTK     411    A
OUTPTK   *                                                                     *   OUTPTK     412    A
OUTPTK   *                      I - FORMAT PROCESSING                          *   OUTPTK     413    A
OUTPTK   *                                                                     *   OUTPTK     414    A
OUTPTK   *                CONVERT THE BINARY INTEGER DATA ITEM TO DECIMAL      *   OUTPTK     415    A
OUTPTK   *             REPRESENTATION, RIGHT-JUSTIFIED WITHIN THE SPECIFIED    *   OUTPTK     416    A
OUTPTK   *             FIELD (WITH BLANK FILL IF NECESSARY).  AFTER THE DECIMAL *   OUTPTK     417    A
OUTPTK   *             FORM HAS BEEN OBTAINED GO TO THE A-FORMAT PROCESSING    *   OUTPTK     418    A
OUTPTK   *             SECTION TO INSERT THE CHARACTERS INTO THE OUTPUT LINE.  *   OUTPTK     419    A
OUTPTK   *                                                                     *   OUTPTK     420    A
OUTPTK   ***********************************************************************   OUTPTK     421    A
OUTPTK            SPACE  1                                                          OUTPTK     422    A
OUTPTK   IFORMCON DATA   10H          ,10.0P0,0.100000000001P48                     OUTPTK     423    A
OUTPTK            SPACE  1                                                          OUTPTK     424    A
OUTPTK   IFORM    FORM                                                              OUTPTK     425    A
OUTPTK            SB5    X6             IF THE FIELD WIDTH IS LESS THAN ELEVEN      OUTPTK     426    A
OUTPTK            LE     B5,B6,IFORM.1    CHARACTERS THEN GO TO 'IFORM.1',          OUTPTK     427    A
OUTPTK            SPACE  1                                                          OUTPTK     428    A
OUTPTK            SX6    B5-B6          OTHERWISE CALL 'SPACE' TO SPACE THE         OUTPTK     429    A
OUTPTK            CALL   SPACE            EXCESS NUMBER OF POSITIONS (I.E. IF       OUTPTK     430    A
OUTPTK            SB5    B6               THE FIELD WIDTH IS TWELVE CHARACTERS      OUTPTK     431    A
OUTPTK                                  THEN SPACE TWO CHARACTERS).                 OUTPTK     432    A
OUTPTK            SPACE  1                                                          OUTPTK     433    A
OUTPTK    IFORM.1 BSS    0                                                          OUTPTK     434    A
OUTPTK            PL     X1,IFORM.12    IF NEGATIVE NUMBER COMPLEMENT AND           OUTPTK     435    A
OUTPTK            BX1    -X1              CHANGE TO POSITIVE NUMBER                 OUTPTK     436    A
OUTPTK            SB6    B0               AND SET FLAG                              OUTPTK     437    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OUTPTK

```
OUTPTK    IFORM.12 BSS    0                                                    OUTPTK     438    A
OUTPTK             PX1    X1             CONVERT THE INTEGER TO FLOATING       OUTPTK     439    A
OUTPTK             SB7    B5               POINT FORM (UN-NORMALIZED) AND FETCH OUTPTK     440    A
OUTPTK             SA5    IFORMCON         THE CONVERSION CONSTANTS.           OUTPTK     441    A
OUTPTK             BX7    X5                                                   OUTPTK     442    A
OUTPTK             SA4    A5+B1                                                OUTPTK     443    A
OUTPTK             SA5    A4+B1                                                OUTPTK     444    A
OUTPTK             SPACE  1                                                    OUTPTK     445    A
OUTPTK   IFORM.2   DX6    X5*X1          CONVERT THE BINARY VALUE, FORMING THE OUTPTK     446    A
OUTPTK             FX1    X5*X1            DECIMAL REPRESENTATION A DIGIT AT A OUTPTK     447    A
OUTPTK             SB5    B5-6             TIME (FROM RIGHT TO LEFT).          OUTPTK     448    A
OUTPTK             FX6    X4*X6            METHOD: LET 'B' BE THE BINARY VALUE OUTPTK     449    A
OUTPTK             SX6    X6+1R0-1R          1)  XD = B*0.1   (FRACTIONAL PART) OUTPTK     450    A
OUTPTK             IX7    X6+X7              2)  B  = B*0.1   (INTEGRAL PART)   OUTPTK     451    A
OUTPTK             NX6    X1                 3)  D  = XD*10.0 (DECIMAL DIGIT)   OUTPTK     452    A
OUTPTK             LX7    -6                                                   OUTPTK     453    A
OUTPTK             NZ     X6,IFORM.2                                           OUTPTK     454    A
OUTPTK             NE     B6,B0,IFORM.3                                        OUTPTK     455    A
OUTPTK             SX5    1R--1R                                               OUTPTK     456    A
OUTPTK             SB6    60               RESTORE B6                          OUTPTK     457    A
OUTPTK             IX7    X7+X5                                                OUTPTK     458    A
OUTPTK    IFORM.3  BSS    0                                                    OUTPTK     459    A
OUTPTK             SPACE  1                                                    OUTPTK     460    A
OUTPTK             SX6    B7             CALL 'AFORM' TO APPEND THE PROPERLY   OUTPTK     461    A
OUTPTK             SB5    B6-B5            POSITIONED AND JUSTIFIED DIGITS.    OUTPTK     462    A
OUTPTK             LX1    X7,B5                                                OUTPTK     463    A
OUTPTK             ZR     AFORM                                                OUTPTK     464    A
OUTPTK             TITLE  L - FORMAT PROCESSING SECTION                        OUTPTK     465    A
OUTPTK   ********************************************************************OUTPTK     466    A
OUTPTK   *        L - FORMAT PROCESSING                                     *OUTPTK     467    A
OUTPTK   *        PRINTS A T IF THE LOGICAL VARIABLE HAS A VALUE OF TRUE, ELSE *OUTPTK     468    A
OUTPTK   *        PRINTS A F IF THE LOGICAL VARIABLE HAS A VALUE OF FALSE   *OUTPTK     469    A
OUTPTK   ********************************************************************OUTPTK     470    A
OUTPTK    LFORM    FORM                                                       OUTPTK     471    A
OUTPTK             SB5    X6             IF THE FIELD WIDTH IS LESS THAN       OUTPTK     472    A
OUTPTK             LE     B5,B6,LFORM.1    ELEVEN CHARACTERS THEN GO TO LFORM.1OUTPTK     473    A
OUTPTK                                                                         OUTPTK     474    A
OUTPTK             SX6    B5-B6            OTHERWISE CALL SPACE TO SPACE THE   OUTPTK     475    A
OUTPTK             CALL   SPACE            EXCESS NUMBER OF POSITIONS          OUTPTK     476    A
OUTPTK             SB5    B6                                                   OUTPTK     477    A
OUTPTK                                                                         OUTPTK     478    A
OUTPTK    LFORM.1  BSS    0                                                    OUTPTK     479    A
OUTPTK             SA5    IFORMCON         FILL WORD WITH BLANKS               OUTPTK     480    A
OUTPTK             SX6    1RF-1R           F                                   OUTPTK     481    A
OUTPTK             PL     X1,LFORM.2                                           OUTPTK     482    A
OUTPTK             SX6    1RT-1R           T                                   OUTPTK     483    A
OUTPTK    LFORM.2  BSS    0                                                    OUTPTK     484    A
OUTPTK             SB7    B6-B5                                                OUTPTK     485    A
OUTPTK             IX1    X6+X5                                                OUTPTK     486    A
OUTPTK             LX1    X1,B7            SHIFT LETTER TO BOTTOM OF FIELD     OUTPTK     487    A
OUTPTK             SX6    B5                                                   OUTPTK     488    A
OUTPTK             EQ     AFORM                                                OUTPTK     489    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OUTPTK

```
OUTPTK            TITLE                 O - FORMAT PROCESSING SECTION                OUTPTK     490    A
OUTPTK   ***********************************************************************    OUTPTK     491    A
OUTPTK   *                                                                     *    OUTPTK     492    A
OUTPTK   *                       O - FORMAT PROCESSING                         *    OUTPTK     493    A
OUTPTK   *                                                                     *    OUTPTK     494    A
OUTPTK   *                CONVERT THE BINARY DATA ITEM TO OCTAL FORM (ONLY THE *    OUTPTK     495    A
OUTPTK   *              NUMBER OF DIGITS SPECIFIED THOUGH), BLANK FILLING IF   *    OUTPTK     496    A
OUTPTK   *              NECESSARY.                                             *    OUTPTK     497    A
OUTPTK   *                                                                     *    OUTPTK     498    A
OUTPTK   ***********************************************************************    OUTPTK     499    A
OUTPTK            SPACE  1                                                           OUTPTK     500    A
OUTPTK   OFORMCON DATA   10H0000000000,0                                             OUTPTK     501    A
OUTPTK            SPACE  1                                                           OUTPTK     502    A
OUTPTK    ZFORM   SA4    =1H                                                         OUTPTK     503    A
OUTPTK            SX7    1R0-1R                                                      OUTPTK     504    A
OUTPTK            EQ     OFORM1                                                      OUTPTK     505    A
OUTPTK                                                                               OUTPTK     506    A
OUTPTK    OFORM   SA4    =10H0000000000                                              OUTPTK     507    A
OUTPTK            MX7    0                                                           OUTPTK     508    A
OUTPTK                                                                               OUTPTK     509    A
OUTPTK    OFORM1  SA7    OFORMCON+1                                                  OUTPTK     510    A
OUTPTK            BX7    X4                                                          OUTPTK     511    A
OUTPTK            SA7    A7-B1                                                       OUTPTK     512    A
OUTPTK            SB7    X6                                                          OUTPTK     513    A
OUTPTK            SB5    B6+B6                                                       OUTPTK     514    A
OUTPTK            LE     B7,B6,OFORM.3   IF THE WIDTH OF THE OCTAL FORMAT IS         OUTPTK     515    A
OUTPTK                                      LESS THAN ELEVEN CHARACTERS THEN         OUTPTK     516    A
OUTPTK                                      GO TO 'OFORM.4'.                         OUTPTK     517    A
OUTPTK            SPACE  1                                                           OUTPTK     518    A
OUTPTK            LE     B7,B5,OFORM.1   IF THE WIDTH IS BETWEEN ELEVEN AND           OUTPTK     519    A
OUTPTK                                      TWENTY CHARACTERS THEN GO TO             OUTPTK     520    A
OUTPTK                                      'OFORM.1',                               OUTPTK     521    A
OUTPTK            SX6    B7-B5           OTHERWISE CALL 'SPACE' TO SKIP OVER         OUTPTK     522    A
OUTPTK            CALL   SPACE              THE EXCESS POSITIONS AND THEN            OUTPTK     523    A
OUTPTK            SX6    B6+B6              PROCESS AS A TWENTY DIGIT CONVERSION.    OUTPTK     524    A
OUTPTK            SB7    B6+B6                                                       OUTPTK     525    A
OUTPTK            SPACE  1                                                           OUTPTK     526    A
OUTPTK   OFORM.1  SB5    B7-B6           CONVERT THE BINARY VALUE TO TWENTY          OUTPTK     527    A
OUTPTK            MX7    60-3               OCTAL DIGITS, EXTRACTING TWO DIGITS      OUTPTK     528    A
OUTPTK            BX4    X7                 AT A TIME.                               OUTPTK     529    A
OUTPTK            LX7    30                                                          OUTPTK     530    A
OUTPTK            BX4    X7*X4                                                       OUTPTK     531    A
OUTPTK            IX3    X3-X3                                                       OUTPTK     532    A
OUTPTK            MX5    0                                                           OUTPTK     533    A
OUTPTK            SB7    B6                                                          OUTPTK     534    A
OUTPTK            SPACE  1                                                           OUTPTK     535    A
OUTPTK   OFORM.2  BX7    -X4*X1          EXTRACT TWO OCTAL DIGITS (THIRTY BITS       OUTPTK     536    A
OUTPTK            AX1    3                  APART),                                  OUTPTK     537    A
OUTPTK            SX6    X7+1R0             CONVERT THE DIGITS TO DISPLAY CODE       OUTPTK     538    A
OUTPTK            IX3    X3+X6              AND APPEND THEM TO THE ACCUMULATED       OUTPTK     539    A
OUTPTK            AX7    30                 DIGITS.                                  OUTPTK     540    A
OUTPTK            SX6    X7+1R0                                                      OUTPTK     541    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OUTPTK

```
OUTPTK          IX5    X5+X6                                                OUTPTK   542   A
OUTPTK          LX3    -6                                                   OUTPTK   543   A
OUTPTK          SB7    B7-6                                                 OUTPTK   544   A
OUTPTK          LX5    -6                                                   OUTPTK   545   A
OUTPTK          GT     B7,B0,OFORM.2   CONTINUE UNTIL ALL DIGITS ARE PROCESSED  OUTPTK   546   A
OUTPTK          SPACE  1                                                    OUTPTK   547   A
OUTPTK          BX1    X5              CALL 'AFORM' TO PLACE THE APPROPRIATE  OUTPTK   548   A
OUTPTK          SX6    B5                DIGITS INTO THE OUTPUT LINE.       OUTPTK   549   A
OUTPTK          INVOKE RFORM                                                OUTPTK   550   A
OUTPTK          BX1    X3                                                   OUTPTK   551   A
OUTPTK          SX6    B6                                                   OUTPTK   552   A
OUTPTK          INVOKE AFORM                                                OUTPTK   553   A
OUTPTK          ZR     NEXTELEM        RETURN FOR THE NEXT ELEMENT.         OUTPTK   554   A
OUTPTK          SPACE  2                                                    OUTPTK   555   A
OUTPTK   OFORM.3 AX6   1               IF TEN OR FEWER DIGITS ARE TO BE     OUTPTK   556   A
OUTPTK          SB5    X6+B1             CONVERTED THEN PERFORM THE CONVERSION  OUTPTK   557   A
OUTPTK          SA5    OFORMCON          ONE DIGIT AT A TIME, FROM RIGHT TO   OUTPTK   558   A
OUTPTK          MX7    1                 LEFT.  LEADING ZERO DIGITS WILL BE   OUTPTK   559   A
OUTPTK          SB5    B6-B5             REPLACED BY THE CHARACTER WHICH IS   OUTPTK   560   A
OUTPTK          SA4    A5+B1             REPLICATED TEN TIMES IN 'OFORMCON'   OUTPTK   561   A
OUTPTK          AX7    X7,B5             (AND OF COURSE THIS MAY BE THE       OUTPTK   562   A
OUTPTK          BX1    -X7*X1            CHARACTER ZERO).  SEE THE DESCRIPTION  OUTPTK   563   A
OUTPTK          SB5    X4                ABOVE FOR FUTHER DETAILS.            OUTPTK   564   A
OUTPTK          SX3    B7                                                   OUTPTK   565   A
OUTPTK          MX4    60-3                                                 OUTPTK   566   A
OUTPTK          SPACE  1                                                    OUTPTK   567   A
OUTPTK   OFORM.4 BX6   -X4*X1          EXTRACT ONE DIGIT.                   OUTPTK   568   A
OUTPTK          AX1    3               SHIFT VALUE TO POSITION THE NEXT DIGIT.  OUTPTK   569   A
OUTPTK          SX7    X6+B5           CONVERT THE DIGIT TO THE VALUE WHICH,  OUTPTK   570   A
OUTPTK          IX5    X5+X7             WHEN ADDED TO 'OFORMCON', WILL BE THE  OUTPTK   571   A
OUTPTK          SB7    B7-6              DISPLAY CODE FOR THE DIGIT.        OUTPTK   572   A
OUTPTK          NO                       DECREMENT WIDTH.                   OUTPTK   573   A
OUTPTK          LX5    -6              POSITION THE OCNVERTED DIGITS.       OUTPTK   574   A
OUTPTK          NZ     X1,OFORM.4      CONTINUE IF ANY NON-ZERO DIGITS REMAIN.  OUTPTK   575   A
OUTPTK          SPACE  1                                                    OUTPTK   576   A
OUTPTK          SB5    B6-B7           POSITION THE CONVERTED DIGITS AND    OUTPTK   577   A
OUTPTK          LX1    X5,B5             INVOKE 'AFORM' TO APPEND THE DIGITS  OUTPTK   578   A
OUTPTK          SX6    X3                TO THE OUTPUT LINE.                OUTPTK   579   A
OUTPTK          INVOKE AFORM                                                OUTPTK   580   A
OUTPTK          ZR     NEXTELEM        RETURN FOR THE NEXT ELEMENT.         OUTPTK   581   A
OUTPTK          TITLE                  H - FORMAT PROCESSING SECITON         OUTPTK   582   A
OUTPTK   ************************************************************       OUTPTK   583   A
OUTPTK   *                                                          *       OUTPTK   584   A
OUTPTK   *                                                          *       OUTPTK   585   A
OUTPTK   *                     H - FORMAT PROCESSING                *       OUTPTK   586   A
OUTPTK   *                                                          *       OUTPTK   587   A
OUTPTK   *                                                          *       OUTPTK   588   A
OUTPTK   ************************************************************       OUTPTK   589   A
OUTPTK          SPACE  1                                                    OUTPTK   590   A
OUTPTK   HFORM  SB7    X6                                                   OUTPTK   591   A
OUTPTK          SB5    B2+6                                                 OUTPTK   592   A
OUTPTK          LE     B7,B5,HFORM.2   IF THE CURRENT FORMAT WORD CONTAINS  OUTPTK   593   A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OUTPTK

```
OUTPTK                                  ALL OF THE HOLLERITH TEXT THEN GO          OUTPTK     594    A
OUTPTK                                  TO 'HFORM.2'.                              OUTPTK     595    A
OUTPTK            SPACE  1                                                         OUTPTK     596    A
OUTPTK   HFORM.1  SX7    B7-B5          POSITION THE HOLLERITH TEXT IN A-FORM      OUTPTK     597    A
OUTPTK            SB7    B6-B5            FOR PROCESSING BY THE 'AFORM' SECTION    OUTPTK     598    A
OUTPTK            LX1    X2,B7            (ALSO SET THE LENGTH OF THE TEXT         OUTPTK     599    A
OUTPTK                                    PORTION FOR 'HFORM').                    OUTPTK     600    A
OUTPTK            SPACE  1                                                         OUTPTK     601    A
OUTPTK            SX6    B5             SET THE LENGTH OF THE CHARACTERS TO BE     OUTPTK     602    A
OUTPTK                                    PROCESSED BY THE 'AFORM' SECTION.        OUTPTK     603    A
OUTPTK            SPACE  1                                                         OUTPTK     604    A
OUTPTK            SA7    HCOUNT         SAVE THE LENGTH OF THE HOLLERITH STRING    OUTPTK     605    A
OUTPTK                                    WHICH REMAINS TO BE TRANSMITTED.         OUTPTK     606    A
OUTPTK            SPACE  1                                                         OUTPTK     607    A
OUTPTK            INVOKE AFORM          INVOKE THE 'AFORM' PROCESSING SECTION,     OUTPTK     608    A
OUTPTK                                    BUT RETURN HERE TO COMPLETE THE          OUTPTK     609    A
OUTPTK                                    H-FORMAT PROCESSING.                     OUTPTK     610    A
OUTPTK            SPACE  1                                                         OUTPTK     611    A
OUTPTK            SA2    A2+B1          FETCH THE NEXT FORMAT WORD AND SET THE     OUTPTK     612    A
OUTPTK            SB5    B6               AVAILABLE STRING LENGTH TO SIXTY BITS    OUTPTK     613    A
OUTPTK            SPACE  1                                                         OUTPTK     614    A
OUTPTK            SA5    HCOUNT         LOAD THE LENGTH OF THE REMAINING           OUTPTK     615    A
OUTPTK            SB7    X5               HOLLERITH TEXT TO BE TRANSMITTED AND     OUTPTK     616    A
OUTPTK            GT     B7,B5,HFORM.1    IF THIS FORMAT WORD WILL NOT             OUTPTK     617    A
OUTPTK                                    COMPLETELY SATISFY THE TRANSFER TEHN     OUTPTK     618    A
OUTPTK                                    GO TO 'HFORM.1'.                         OUTPTK     619    A
OUTPTK            SPACE  1                                                         OUTPTK     620    A
OUTPTK   HFORM.2  SB2    B5-B7          RESET THE FORMAT SHIFT COUNT,              OUTPTK     621    A
OUTPTK            SB5    B6-B5            POSITION THE LAST TEXT SECTION (AND      OUTPTK     622    A
OUTPTK            LX1    X2,B5          THE LENGTH OF THIS SECTION) AND            OUTPTK     623    A
OUTPTK            SX6    B7             INVOKE THE 'AFORM' PROCESSING.             OUTPTK     624    A
OUTPTK            SB2    B2-6                                                      OUTPTK     625    A
OUTPTK            INVOKE AFORM                                                     OUTPTK     626    A
OUTPTK            SPACE  1                                                         OUTPTK     627    A
OUTPTK            PL     B2,NEXTDESC    IF THE CURRENT FORMAT WORD HAS BEEN        OUTPTK     628    A
OUTPTK            SA2    A2+B1            EXHAUSTED THEN FETCH THE NEXT WORD       OUTPTK     629    A
OUTPTK            SB2    B6+B2            AND ADJUST THE FORMAT POINTERS BEFORE    OUTPTK     630    A
OUTPTK            ZR     NEXTDESC         RETURNING TO 'NEXTDESC'.                 OUTPTK     631    A
OUTPTK            TITLE                 DELIM -- HOLLERITH STRING PROCESSING (D     OUTPTK     632    A
OUTPTK   ,ELIMITER FORM)                                                           OUTPTK     633    A
OUTPTK   ************************************************************************  OUTPTK     634    A
OUTPTK   *                                                                      *  OUTPTK     635    A
OUTPTK   *                                                                      *  OUTPTK     636    A
OUTPTK   *                HOLLERITH STRING PROCESSING (DELIMITER FORM)          *  OUTPTK     637    A
OUTPTK   *                                                                      *  OUTPTK     638    A
OUTPTK   *                                                                      *  OUTPTK     639    A
OUTPTK   ************************************************************************  OUTPTK     640    A
OUTPTK            SPACE  1                                                         OUTPTK     641    A
OUTPTK   DELIM    SX3    B5             PLACE THE STRING DELIMITER INTO            OUTPTK     642    A
OUTPTK            SB5    6                REGISTER X5 AND 6 INTO REGISTER          OUTPTK     643    A
OUTPTK                                    B5 FOR THE DURATION OF STRING            OUTPTK     644    A
OUTPTK                                    PROCESSING.                              OUTPTK     645    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OUTPTK

```
OUTPTK            SPACE  1                                                    OUTPTK    646    A
OUTPTK   DELIM.1  SB7    B6              INITIALIZE FOR ACCUMULATING UP TO TEN OUTPTK    647    A
OUTPTK            BX1    X1-X1             HOLLERITH CHARACTERS AT A TIME BEFORE OUTPTK    648    A
OUTPTK            MX7    60-6              INVOKING THE 'AFORM' PROCESSING.      OUTPTK    649    A
OUTPTK            SPACE  1                                                    OUTPTK    650    A
OUTPTK   DELIM.2  AX4    X2,B2           EXTRACT THE NEXT CHARACTER FROM THE   OUTPTK    651    A
OUTPTK            SB2    B2-B5             FORMAT, UPDATE THE FORMAT POINTERS,   OUTPTK    652    A
OUTPTK            BX5    -X7*X4            AND IF THE END OF THE HOLLERITH       OUTPTK    653    A
OUTPTK            IX6    X5-X3             STRING HAS BEEN REACHED THEN GO TO    OUTPTK    654    A
OUTPTK            PL     B2,DELIM.3        'DELIM.4'.                            OUTPTK    655    A
OUTPTK            SB2    B2+B6             (RESET FORMAT POINTERS WHEN STEPPING  OUTPTK    656    A
OUTPTK            SA2    A2+B1             OVER A WORD BOUNDARY.)                OUTPTK    657    A
OUTPTK   DELIM.3  ZR     X6,DELIM.4                                            OUTPTK    658    A
OUTPTK            SPACE  1                                                    OUTPTK    659    A
OUTPTK            BX1    X5+X1           OTHERWISE APPEND THE CHARACTER TO THE OUTPTK    660    A
OUTPTK            SB7    B7-B5             ACCUMULATED STRING AND GO TO          OUTPTK    661    A
OUTPTK            LX1    6                 'DELIM.2' IF TEN CHARACTERS HAVE NOT  OUTPTK    662    A
OUTPTK            GT     B7,B0,DELIM.2     YET BEEN ACCUMULATED.                 OUTPTK    663    A
OUTPTK            SX6    B6              SET THE 'AFORM' COUNT TO TEN CHARACTERS OUTPTK    664    A
OUTPTK            LX1    60-6              AND INVOKE THE 'AFORM' PROCESSING.    OUTPTK    665    A
OUTPTK            INVOKE AFORM                                                 OUTPTK    666    A
OUTPTK            SPACE  1                                                    OUTPTK    667    A
OUTPTK            SB5    6                                                    OUTPTK    668    A
OUTPTK            ZR     DELIM.1         REENTER THE CYCLE AT 'DELIM.1' TO     OUTPTK    669    A
OUTPTK                                     INITIALIZE.                         OUTPTK    670    A
OUTPTK            SPACE  1                                                    OUTPTK    671    A
OUTPTK   DELIM.4  SX6    B6-B7           SET THE COUNT OF ACCUMULATED CHARACTERS OUTPTK    672    A
OUTPTK            SB7    B7-B5                                                 OUTPTK    673    A
OUTPTK            LX1    X1,B7             AND INVOKE THE 'AFORM' PROCESSING IF  OUTPTK    674    A
OUTPTK            ZR     X6,NEXTELEM       THERE WERE ANY ACCUMULATED CHARACTERS OUTPTK    675    A
OUTPTK            INVOKE AFORM                                                 OUTPTK    676    A
OUTPTK            SPACE  1                                                    OUTPTK    677    A
OUTPTK            ZR     NEXTDESC        AFTER PROCESSING THE CHARACTERS (IF   OUTPTK    678    A
OUTPTK                                     ANY) RETURN TO 'NEXTDESC' FOR THE   OUTPTK    679    A
OUTPTK                                     THE NEXT FORMAT FIELD DESCRIPTOR.   OUTPTK    680    A
OUTPTK            TITLE                 LEFT AND RIGHT PARENTHESIS PROCESSING S OUTPTK    681    A
OUTPTK   ,ECTION                                                              OUTPTK    682    A
OUTPTK   ************************************************************************ OUTPTK    683    A
OUTPTK   *                                                                    *  OUTPTK    684    A
OUTPTK   *                      LEFT PARENTHESIS PROCESSING                   *  OUTPTK    685    A
OUTPTK   *                                                                    *  OUTPTK    686    A
OUTPTK   *               SAVE THE PREVIOUS GROUP INFORMATION (BEGINNING GROUP *  OUTPTK    687    A
OUTPTK   *            ADDRESS AND GROUP REPEAT COUNT) IN THE LEVEL STACK AND  *  OUTPTK    688    A
OUTPTK   *            SET THE CURRENT BEGINNING GROUP ADDRESS AND GROUP REPEAT *  OUTPTK    689    A
OUTPTK   *            COUNT.  (IF THIS IS A LEVEL-1 GROUP, THEN ALSO SAVE THE *  OUTPTK    690    A
OUTPTK   *            CURRENT INFORMATION FOR RESTART PURPOSES WHEN THE END OF *  OUTPTK    691    A
OUTPTK   *            THE FORMAT IS REACHED.)                                 *  OUTPTK    692    A
OUTPTK   *                                                                    *  OUTPTK    693    A
OUTPTK   ************************************************************************ OUTPTK    694    A
OUTPTK            SPACE  1                                                    OUTPTK    695    A
OUTPTK   LPFORM   SA3    GROUPADR                                             OUTPTK    696    A
OUTPTK            SA5    GRC                                                  OUTPTK    697    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OUTPTK

```
OUTPTK          SX4    B2                                                  OUTPTK     698    A
OUTPTK          LX3    18                                                  OUTPTK     699    A
OUTPTK          SX6    A2                                                  OUTPTK     700    A
OUTPTK          LX4    18                                                  OUTPTK     701    A
OUTPTK          BX7    X4+X6                                               OUTPTK     702    A
OUTPTK          SA4    LEVEL          (THE LEVEL NUMBER STORED IN 'LEVEL' IS  OUTPTK  703    A
OUTPTK                                   ONE LESS THAN THE GROUP LEVEL.)   OUTPTK     704    A
OUTPTK          SPACE  1                                                   OUTPTK     705    A
OUTPTK          SA7    A3             SET THE BEGINNING ADDRESS OF THE    OUTPTK     706    A
OUTPTK                                   CURRENT FORMAT GROUP.             OUTPTK     707    A
OUTPTK          BX6    X3+X5                                               OUTPTK     708    A
OUTPTK          LX3    X7                                                  OUTPTK     709    A
OUTPTK          SA5    R                                                   OUTPTK     710    A
OUTPTK          SA6    LEVSTACK+1+X4  SAVE THE PREVIOUS GROUP INFORMATION. OUTPTK     711    A
OUTPTK          BX7    X5                                                  OUTPTK     712    A
OUTPTK          SX6    X4+B1                                               OUTPTK     713    A
OUTPTK          SA7    GRC            SET THE CURRENT GROUP REPEAT COUNT.  OUTPTK     714    A
OUTPTK          SB7    X6                                                  OUTPTK     715    A
OUTPTK          SA6    A4             INCREMENT THE CURRENT LEVEL NUMBER.  OUTPTK     716    A
OUTPTK          SPACE  1                                                   OUTPTK     717    A
OUTPTK          NE     B7,B1,NEXTDESC IF THE NEW GROUP LEVEL IS NOT LEVEL ONE OUTPTK  718    A
OUTPTK                                   THEN GO TO 'NEXTDESC',            OUTPTK     719    A
OUTPTK          SPACE  1                                                   OUTPTK     720    A
OUTPTK          LX3    18             OTHERWISE SAVE THE NEW GROUP REPEAT OUTPTK     721    A
OUTPTK          BX7    X3+X7             COUNT AND BEGINNING ADDRESS FOR   OUTPTK     722    A
OUTPTK          LX6    3*18              POSSIBLE USE AS A RESTART POSITION OUTPTK    723    A
OUTPTK          IX7    X6+X7             WHEN THE END OF THE FORMAT STRING IS OUTPTK  724    A
OUTPTK          SA7    RESCAN            REACHED                           OUTPTK     725    A
OUTPTK          EQ     NEXTDESC                                            OUTPTK     726    A
OUTPTK          EJECT                                                      OUTPTK     727    A
OUTPTK   ***********************************************************      OUTPTK     728    A
OUTPTK   *                                                         *      OUTPTK     729    A
OUTPTK   *                 RIGHT PARENTHESIS PROCESSING            *      OUTPTK     730    A
OUTPTK   *                                                         *      OUTPTK     731    A
OUTPTK   *                                                         *      OUTPTK     732    A
OUTPTK   ***********************************************************      OUTPTK     733    A
OUTPTK          SPACE  1                                                   OUTPTK     734    A
OUTPTK   RPFORM SA5    GRC                                                 OUTPTK     735    A
OUTPTK          MX7    59                                                  OUTPTK     736    A
OUTPTK          IX6    X5+X7                                               OUTPTK     737    A
OUTPTK          ZR     X6,RPFORM.1    IF THE GROUP REPEAT COUNT IS EXHAUSTED OUTPTK  738    A
OUTPTK                                   THEN GO TO 'RPFORM.1'.            OUTPTK     739    A
OUTPTK          SPACE  1                                                   OUTPTK     740    A
OUTPTK          SA4    GROUPADR       OTHERWISE RESET THE FORMAT POINTERS TO OUTPTK  741    A
OUTPTK          SA2    X4                THE BEGINNING OF THE CURRENT GROUP OUTPTK     742    A
OUTPTK          AX4    18                                                  OUTPTK     743    A
OUTPTK          SA6    A5                                                  OUTPTK     744    A
OUTPTK          SB2    X4                                                  OUTPTK     745    A
OUTPTK          ZR     NEXTDESC       GO TO 'NEXTDESC' FOR HTE NEXT FORMAT OUTPTK     746    A
OUTPTK                                   DESCRIPTOR.                       OUTPTK     747    A
OUTPTK          SPACE  1                                                   OUTPTK     748    A
OUTPTK   RPFORM.1 SA3  LEVEL                                               OUTPTK     749    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OUTPTK

```
OUTPTK            ZR     X3,RPFORM.2     IF THIS IS THE END OF THE FORMAT THEN      OUTPTK   750   A
OUTPTK                                     GO TO 'RPFORM.2'.                        OUTPTK   751   A
OUTPTK            SPACE  2                                                          OUTPTK   752   A
OUTPTK            SA4    LEVSTACK+X3     OTHERWISE RESET THE PARENTHESIS GROUP      OUTPTK   753   A
OUTPTK            IX6    X3+X7             INFORMATION AND DECREMENT THE LEVEL      OUTPTK   754   A
OUTPTK            SX7    X4                                                         OUTPTK   755   A
OUTPTK            SA6    A3                                                         OUTPTK   756   A
OUTPTK            AX4    18                                                         OUTPTK   757   A
OUTPTK            SA7    A5                                                         OUTPTK   758   A
OUTPTK            BX6    X4                                                         OUTPTK   759   A
OUTPTK            SA6    GROUPADR                                                   OUTPTK   760   A
OUTPTK            ZR     NEXTDESC        GO TO 'NEXTDESC' FOR THE NEXT FORMAT       OUTPTK   761   A
OUTPTK                                     DESCRIPTOR.                              OUTPTK   762   A
OUTPTK            SPACE  1                                                          OUTPTK   763   A
OUTPTK    RPFORM.2 SA3     CONTINUE                                                 OUTPTK   764   A
OUTPTK            NZ     X3,RPFORM3  IF EXPLICIT CONTINUATION MODE                  OUTPTK   765   A
OUTPTK            INVOKE SLASH       OUTPUT THE CURRENT LINE                        OUTPTK   766   A
OUTPTK                                                                              OUTPTK   767   A
OUTPTK    RPFORM3 SA3    RESCAN      RESET THE FORMAT POINTERS AND THE CURRENT      OUTPTK   768   A
OUTPTK            SX7    X3                CURRENT GROUP INFORMATION TO RESTART     OUTPTK   769   A
OUTPTK            AX3    18                THE FORMAT SCAN.                         OUTPTK   770   A
OUTPTK            SA2    X3                                                         OUTPTK   771   A
OUTPTK            BX6    X3                                                         OUTPTK   772   A
OUTPTK            SA7    GRC                                                        OUTPTK   773   A
OUTPTK            AX3    18                                                         OUTPTK   774   A
OUTPTK            SA6    GROUPADR                                                   OUTPTK   775   A
OUTPTK            SB2    X3                                                         OUTPTK   776   A
OUTPTK            AX3    18                                                         OUTPTK   777   A
OUTPTK            BX6    X3                                                         OUTPTK   778   A
OUTPTK            SA6    LEVEL                                                      OUTPTK   779   A
OUTPTK            GT     B4,B0,NEXTDESC  IF ANY MORE ITEMS REMAIN TO BE             OUTPTK   780   A
OUTPTK                                     CONVERTED THEN GO TO 'NEXTDESC',         OUTPTK   781   A
OUTPTK            SA7    PARENFLG                                                   OUTPTK   782   A
OUTPTK            ZR     OUTPTCX         OTHERWISE EXIT.                            OUTPTK   783   A
OUTPTK            TITLE                  T - FORMAT PROCESSING SECTION               OUTPTK   784   A
OUTPTK    **      T - FORMAT PROCESSING                                              OUTPTK   785   A
OUTPTK            SPACE  1                                                          OUTPTK   786   A
OUTPTK    TFORM   SA4    OUTADDR         CALCULATE THE CURRENT DISTANCE FROM THE    OUTPTK   787   A
OUTPTK            IX0    X0-X4             BEGINNING OF THE LINE (IN BITS).         OUTPTK   788   A
OUTPTK            LX7    X0,B1                                                      OUTPTK   789   A
OUTPTK            SA5    =60.0                                                      OUTPTK   790   A
OUTPTK            LX0    5                                                          OUTPTK   791   A
OUTPTK            IX7    X0-X7                                                      OUTPTK   792   A
OUTPTK            LX0    X7,B1                                                      OUTPTK   793   A
OUTPTK            SB3    X0+B3             (CURRENT DISTANCE.)                      OUTPTK   794   A
OUTPTK            SA4    MAXCOUNT                                                   OUTPTK   795   A
OUTPTK            SB5    X6                (POSITION TO TABULATE TO, IN BITS.)      OUTPTK   796   A
OUTPTK            BX3    X3-X3                                                      OUTPTK   797   A
OUTPTK            GT     B5,B3,TFORM.2   IF A TAB FORWARD THEN GO TO 'TFORM.2'.     OUTPTK   798   A
OUTPTK            SPACE  1                                                          OUTPTK   799   A
OUTPTK            SB7    X4              SINCE THIS IS A TAB BACKWARD, SET          OUTPTK   800   A
OUTPTK            LE     B3,B7,TFORM.1     'MAXCOUNT' TO THE MAXIMUM OF THE         OUTPTK   801   A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OUTPTK

```
OUTPTK            SA6    A4                PREVIOUS VALUE OF 'MAXCOUNT' AND THE        OUTPTK   802   A
OUTPTK                                     CURRENT DISTANCE TO THE BEGINNING OF        OUTPTK   803   A
OUTPTK                                     THE LINE.                                   OUTPTK   804   A
OUTPTK            SPACE  1                                                             OUTPTK   805   A
OUTPTK   TFORM.1  PX4    X6              RESET THE LINE POINTERS TO REFLECT            OUTPTK   806   A
OUTPTK            FX7    X4/X5             THE POSITION TABULATED TO.                  OUTPTK   807   A
OUTPTK            SA4    OUTADDR                                                       OUTPTK   808   A
OUTPTK            UX5    X7,B7                                                         OUTPTK   809   A
OUTPTK            LX7    X5,B7                                                         OUTPTK   810   A
OUTPTK            IX0    X7+X4                                                         OUTPTK   811   A
OUTPTK            LX5    X7,B1                                                         OUTPTK   812   A
OUTPTK            IX4    X5+X5                                                         OUTPTK   813   A
OUTPTK            LX7    6                                                             OUTPTK   814   A
OUTPTK            IX4    X7-X4                                                         OUTPTK   815   A
OUTPTK            SB7    X4                                                            OUTPTK   816   A
OUTPTK            SB3    B5-B7                                                         OUTPTK   817   A
OUTPTK            ZR     X3,NEXTDESC     IF NO SPACING TO BE DONE THEN RETURN          OUTPTK   818   A
OUTPTK            BX6    X3              OTHERWISE CALL 'SPACE' TO COMPLETE THE        OUTPTK   819   A
OUTPTK            CALL   SPACE             TABULATION.                                 OUTPTK   820   A
OUTPTK            ZR     NEXTDESC        RETURN FOR THE NEXT FORMAT DESCRIPTOR.        OUTPTK   821   A
OUTPTK            SPACE  1                                                             OUTPTK   822   A
OUTPTK   TFORM.2  SB7    X4              IF THIS TAB FORWARD DOES NOT EXCEED THE       OUTPTK   823   A
OUTPTK            LE     B3,B7,TFORM.3     MAXIMUM PREVIOUS LINE LENGTH ATTAINED       OUTPTK   824   A
OUTPTK            SB7    B3                THEN GO TO 'IFORM.1' TO RESET THE           OUTPTK   825   A
OUTPTK    TFORM.3 LE     B5,B7,TFORM.1     LINE POINTERS.                              OUTPTK   826   A
OUTPTK            SPACE  1                                                             OUTPTK   827   A
OUTPTK            SX3    B5-B7           OTHERWISE PERFORM THE TAB BY HAVING THE       OUTPTK   828   A
OUTPTK            BX6    X4                LINE POINTERS RESET TO THE MAXIMUM          OUTPTK   829   A
OUTPTK            SB5    X4                LINE LENGTH AND THEN SPACING OUT TO         OUTPTK   830   A
OUTPTK            EQ     TFORM.1           THE POSITION TO BE TABULATED TO             OUTPTK   831   A
OUTPTK                                                                                OUTPTK   832   A
OUTPTK   **       X - FORMAT PROCESSING                                               OUTPTK   833   A
OUTPTK            SPACE  1                                                             OUTPTK   834   A
OUTPTK   XFORM    FORM                                                                OUTPTK   835   A
OUTPTK            CALL   SPACE           CALL 'SPACE' TO SPACE THE SPECIFIED          OUTPTK   836   A
OUTPTK            ZR     NEXTDESC          NUMBER OF POSITIONS AND THE RETURN.        OUTPTK   837   A
OUTPTK            TITLE                  SLASH -- LINE OUTPUT PROCESSING               OUTPTK   838   A
OUTPTK   ************************************************************************    OUTPTK   839   A
OUTPTK   *                                                                      *     OUTPTK   840   A
OUTPTK   *                                                                      *     OUTPTK   841   A
OUTPTK   *               S L A S H -- LINE OUTPUT PROCESSING                    *     OUTPTK   842   A
OUTPTK   *                                                                      *     OUTPTK   843   A
OUTPTK   *                                                                      *     OUTPTK   844   A
OUTPTK   ************************************************************************    OUTPTK   845   A
OUTPTK            SPACE  1                                                             OUTPTK   846   A
OUTPTK   SLASH    FORM                                                                OUTPTK   847   A
OUTPTK            SX5    A2              SAVE THE CURRENT FORMAT POINTERS, THE         OUTPTK   848   A
OUTPTK            PX6    X5,B2             THE HOLD REGISTER (REGISTER X3),            OUTPTK   849   A
OUTPTK            BX7    X3              AND THE DATA ITEM POINTERS.                   OUTPTK   850   A
OUTPTK            SX4    A1                                                            OUTPTK   851   A
OUTPTK            SA6    RESETFMT                                                      OUTPTK   852   A
OUTPTK            SA7    RESETX3                                                       OUTPTK   853   A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OUTPTK

```
OUTPTK            PX6    X4,B4                                                OUTPTK    854    A
OUTPTK            SB4    8*6                                                  OUTPTK    855    A
OUTPTK            SA5    X0                                                   OUTPTK    856    A
OUTPTK            SA6    RESETITM                                             OUTPTK    857    A
OUTPTK            SA4    OUTADDR                                              OUTPTK    858    A
OUTPTK            IX7    X0-X4          IF THE CURRENT LINE POSITION IS NOT THE OUTPTK    859    A
OUTPTK            SB5    B1+B1            MAXIMUM POSITION ATTAINED THEN RESET OUTPTK    860    A
OUTPTK            LX6    X7,B5            THE LINE POINTERS TO POINT TO THE END OUTPTK    861    A
OUTPTK            LX7    6                                                    OUTPTK    862    A
OUTPTK            SA3    MAXCOUNT         OF THE LINE.                        OUTPTK    863    A
OUTPTK            IX5    X7-X6                                                OUTPTK    864    A
OUTPTK            SX7    X5+B3                                                OUTPTK    865    A
OUTPTK            IX6    X7-X3                                                OUTPTK    866    A
OUTPTK            PX3    X3                                                   OUTPTK    867    A
OUTPTK            PL     X6,SLASH.1     GO TO 'SLASH.1' IF THE CURRENT POSITION OUTPTK    868    A
OUTPTK                                     IS THE FARTHEST.                   OUTPTK    869    A
OUTPTK            SPACE  1                                                    OUTPTK    870    A
OUTPTK            SA5    =0.01666666667  (ONE SIXTIETH)                       OUTPTK    871    A
OUTPTK            FX7    X5*X3            (INTEGRAL PORTION OF BITCOUNT/60)   OUTPTK    872    A
OUTPTK            DX6    X5*X3            (FRACTIONAL PART OF BITCOUNT/60)    OUTPTK    873    A
OUTPTK            SA5    =60.000000001  (SIXTY WITH A ROUND FACTOR.)          OUTPTK    874    A
OUTPTK            IX3    X7+X4            (ADDRESS OF LAST WORD IN LINE)      OUTPTK    875    A
OUTPTK            SX0    X3                                                   OUTPTK    876    A
OUTPTK            FX6    X5*X6            (SET REMAINDER OF BITCOUNT/60)      OUTPTK    877    A
OUTPTK            SB3    X6                                                   OUTPTK    878    A
OUTPTK   SLASH.1  SA3    =8L                                                  OUTPTK    879    A
OUTPTK            SB6    X4             (B6) = LINE FWA                       OUTPTK    880    A
OUTPTK            SA5    X0                                                   OUTPTK    881    A
OUTPTK            LE     B3,B4,SLASH.2  IF THE LAST WORD OF THE LINE CONTAINS OUTPTK    882    A
OUTPTK                                     ENOUGH ROOM FOR THE LINE TERMINATOR OUTPTK    883    A
OUTPTK                                     THEN GO TO 'SLASH.2'.              OUTPTK    884    A
OUTPTK            SPACE  1                                                    OUTPTK    885    A
OUTPTK            MX7    60-6           OTHERWISE FILL OUT THE WORD WITH A    OUTPTK    886    A
OUTPTK            BX5    X7*X5            BLANK AND MOVE THE POINTERS TO THE  OUTPTK    887    A
OUTPTK            SX7    1R               NEXT WORD (INDICATING THAT IT IS    OUTPTK    888    A
OUTPTK            IX6    X5+X7            EMPTY OF COURSE).                   OUTPTK    889    A
OUTPTK            SA5    X0+B1                                                OUTPTK    890    A
OUTPTK            SB3    B0                                                   OUTPTK    891    A
OUTPTK            SA6    X0                                                   OUTPTK    892    A
OUTPTK            SPACE  1                                                    OUTPTK    893    A
OUTPTK   SLASH.2  MX0    1              TERMINATE THE LINE WITH A ZERO BYTE AND OUTPTK    894    A
OUTPTK            SB5    B3-B1            BLANK FILL (IF NECESSARY).          OUTPTK    895    A
OUTPTK            AX0    X0,B5                                                OUTPTK    896    A
OUTPTK            BX7    X0                                                   OUTPTK    897    A
OUTPTK            SB7    B6-B1                                                OUTPTK    898    A
OUTPTK            AX7    59                                                   OUTPTK    899    A
OUTPTK            BX0    X7*X0                                                OUTPTK    900    A
OUTPTK            BX5    X0*X5                                                OUTPTK    901    A
OUTPTK            SB7    A5-B7          (B7) = LINE LENGTH (WORDS)            OUTPTK    902    A
OUTPTK            BX3    -X0*X3                                               OUTPTK    903    A
OUTPTK            IX6    X5+X3                                                OUTPTK    904    A
OUTPTK            SA6    A5             (LAST WORD WITH TERMINATOR.)          OUTPTK    905    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OUTPTK

```
OUTPTK                                                                  OUTPTK     906    A
OUTPTK   *         WRITE LINE TO FILE.                                  OUTPTK     907    A
OUTPTK   *         AT PRESENT, FILE IS RESTRICTED TO -OUTPUT-.          OUTPTK     908    A
OUTPTK                                                                  OUTPTK     909    A
OUTPTK             LISTL  B6,B7          LINE TO SOURCE LISTING FILE   OUTPTK     910    A
OUTPTK                                                                  OUTPTK     911    A
OUTPTK   SLASH.4   SA5    RESETFMT       RESET THE FORMAT POINTERS, THE DATA  OUTPTK  912  A
OUTPTK             UX7    X5,B2            POINTERS, THE SPECIAL REGISTER (X3),  OUTPTK  913  A
OUTPTK             SA2    X5               AND THE REGISTERS WITH CONSTANT VALUE  OUTPTK  914  A
OUTPTK             SA4    RESETITM                                      OUTPTK     915    A
OUTPTK             SA5    TEMPA0.                                       OUTPTK     916    A
OUTPTK             SA0    X5                                            OUTPTK     917    A
OUTPTK             UX7    X4,B4                                         OUTPTK     918    A
OUTPTK             SA1    X4                                            OUTPTK     919    A
OUTPTK             SA3    RESETX3                                       OUTPTK     920    A
OUTPTK             SB1    1                                             OUTPTK     921    A
OUTPTK             SA5    OUTADDR                                       OUTPTK     922    A
OUTPTK             SB6    60                                            OUTPTK     923    A
OUTPTK             BX6    X6-X6                                         OUTPTK     924    A
OUTPTK             SX0    X5                                            OUTPTK     925    A
OUTPTK             SA6    MAXCOUNT                                      OUTPTK     926    A
OUTPTK             SB3    B0                                            OUTPTK     927    A
OUTPTK             ZR     RETURNX        RETURN TO THE CALLER.          OUTPTK     928    A
OUTPTK             TITLE                 FORMAT ITEM-DATA ITEM CONTROL  OUTPTK     929    A
OUTPTK   NEXTELEM  SA5    R              REDUCE THE FIELD DESCRIPTOR REPEAT  OUTPTK  930  A
OUTPTK             MX4    59               COUNT AND ALSO DECREMENT THE COUNT  OUTPTK  931  A
OUTPTK             IX7    X5+X4            OF THE NUMBER OF DATA ITEMS TO BE  OUTPTK  932  A
OUTPTK             SB4    B4-B1            CONVERTED.                    OUTPTK     933    A
OUTPTK             SA7    A5                                            OUTPTK     934    A
OUTPTK             NZ     X7,IFNOMORE    IF THE REPEAT COUNT FOR THE CURRENT  OUTPTK  935  A
OUTPTK                                     FIELD DESCRIPTOR HAS NOT BEEN  OUTPTK  936  A
OUTPTK                                     EXHAUSTED THEN GO TO 'IFNOMORE'.  OUTPTK  937  A
OUTPTK             SPACE  1                                             OUTPTK     938    A
OUTPTK   NEXTDESC  BSS    0              (ENTER HERE FOR NEXT FIELD DESCRIPTOR.)  OUTPTK  939  A
OUTPTK             SPACE  1                                             OUTPTK     940    A
OUTPTK   NEXT.1    AX4    X2,B2          EXTRACT THE NEXT FORMAT CHARACTER (AND  OUTPTK  941  A
OUTPTK             SX6    B1               UPDATE THE FORMAT POINTERS).  OUTPTK     942    A
OUTPTK             SA5    DIGMASK                                       OUTPTK     943    A
OUTPTK             GT     B2,B0,NEXT.2                                  OUTPTK     944    A
OUTPTK             SB2    B6                                            OUTPTK     945    A
OUTPTK             SA2    A2+B1                                         OUTPTK     946    A
OUTPTK   NEXT.2    MX7    60-6           USE THE FUNCTION 'DECIMAL' TO OBTAIN  OUTPTK  947  A
OUTPTK             BX4    -X7*X4           THE BINARY VALUE OF THE REPEAT COUNT  OUTPTK  948  A
OUTPTK             SB5    X4               (ASSUME A VALUE OF ONE IF NO REPEAT  OUTPTK  949  A
OUTPTK             LX3    X5,B5            COUNT IS PRESENT).            OUTPTK     950    A
OUTPTK             SB2    B2-6                                          OUTPTK     951    A
OUTPTK             PL     X3,NEXT.4                                     OUTPTK     952    A
OUTPTK             SX6    X4-1R0                                        OUTPTK     953    A
OUTPTK             RJ     DECIMAL                                       OUTPTK     954    A
OUTPTK             GT     B2,B0,NEXT.3   (ADVANCE BEYOND THE CHARACTER WHICH  OUTPTK  955  A
OUTPTK             SB2    B6               TERMINATED THE DECIMAL CONVERSION.)  OUTPTK  956  A
OUTPTK             SA2    A2+B1                                         OUTPTK     957    A
```

```
         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OUTPTK

```
OUTPTK   NEXT.3    SB2    B2-6                                                  OUTPTK      958    A
OUTPTK             SPACE  1                                                     OUTPTK      959    A
OUTPTK   NEXT.4    SA5    NEEDW          IF THE FORMAT ELEMENT IS A FIELD       OUTPTK      960    A
OUTPTK             SA6    R                DESCRIPTOR AND NEEDS A WIDTH VALUE   OUTPTK      961    A
OUTPTK             SA6    W                (EG. A-FORMAT, I-FORMAT, ETC.) THEN   OUTPTK      962    A
OUTPTK             LX5    X5,B5            USE THE FUNCTION 'DECIMAL' AGAIN TO   OUTPTK      963    A
OUTPTK             LX4    -1               CONVERT THE NEEDED VALUE.             OUTPTK      964    A
OUTPTK             SA3    FORMS+X4                                               OUTPTK      965    A
OUTPTK             MI     X4,NEXT.5        (ALSO STORE AWAY THE ADDRESS OF THE   OUTPTK      966    A
OUTPTK             LX3    30               PROPER PROCESSING SECTION FOR THE     OUTPTK      967    A
OUTPTK   NEXT.5    BX6    X3               FORMAT DESCRIPTOR.)                   OUTPTK      968    A
OUTPTK             SA6    GOCELL                                                 OUTPTK      969    A
OUTPTK             PL     X5,NEXT.6        GO TO 'NEXT.6' IF NO DATA ITEM IS     OUTPTK      970    A
OUTPTK                                      REQUIRED (I.E. NO WIDTH VALUE NEEDED) OUTPTK      971    A
OUTPTK             SPACE  1                                                     OUTPTK      972    A
OUTPTK             SA5    DIGMASK        SETUP DIGIT MASK FOR 'DECIMAL'         OUTPTK      973    A
OUTPTK             SB7    B5                                                     OUTPTK      974    A
OUTPTK             BX6    X6-X6                                                  OUTPTK      975    A
OUTPTK             RJ     DECIMAL                                                OUTPTK      976    A
OUTPTK             SA6    W                                                      OUTPTK      977    A
OUTPTK       SA5 GOCELL                                                          OUTPTK      978    A
OUTPTK       SX6 X5-TFORM                                                        OUTPTK      979    A
OUTPTK       ZR X6,NEXT.6                                                        OUTPTK      980    A
OUTPTK             SPACE  2                                                     OUTPTK      981    A
OUTPTK   IFNOMORE  LE     B4,B0,OUTPTCX  IF NO MORE ITEMS ARE LEFT THEN RETURN. OUTPTK      982    A
OUTPTK             SA1    A1+B1            (ELSE FETCH THE NEXT ITEM.)           OUTPTK      983    A
OUTPTK             SPACE  1                                                     OUTPTK      984    A
OUTPTK   NEXT.6    SA5    W              CALL THE PROPER PROCESSING SECTION     OUTPTK      985    A
OUTPTK             IX6    X5+X5            WITH THE WIDTH VALUE (OR COUNT VALUE  OUTPTK      986    A
OUTPTK             LX4    X6,B1            IN THE CASE OF X-FORMAT, H-FORMAT,    OUTPTK      987    A
OUTPTK             SA5    GOCELL           ETC.) IN BITS IN REGISTER  X6  AND    OUTPTK      988    A
OUTPTK             IX6    X4+X6            THE DATA ITEM, IF ANY, IN REGISTER    OUTPTK      989    A
OUTPTK             SB7    X5               X1.                                   OUTPTK      990    A
OUTPTK             RJ     RETURN                                                 OUTPTK      991    A
OUTPTK             SPACE  2                                                     OUTPTK      992    A
OUTPTK             LT     B3,B6,NEXTELEM UPON RETURN FROM THE PROCESSOR UPDATE   OUTPTK      993    A
OUTPTK             SX0    X0+B1            THE OUTPUT LINE POINTERS IF THE END   OUTPTK      994    A
OUTPTK             SB3    B0               OF A WORD BOUNDARY HAS BEEN REACHED,  OUTPTK      995    A
OUTPTK             ZR     NEXTELEM         THEN GO TO 'NEXTELEM'.                OUTPTK      996    A
OUTPTK             SPACE  3                                                     OUTPTK      997    A
OUTPTK   RETURNX   LT     B3,B6,RETURN   IF THE CURRENT OUTPUT WORD IS NOT FULL OUTPTK      998    A
OUTPTK                                      THEN GO TO 'RETURN'.                OUTPTK      999    A
OUTPTK             SPACE  1                                                     OUTPTK     1000    A
OUTPTK             SX0    X0+B1            INCREMENT THE OUTPUT POINTERS.        OUTPTK     1001    A
OUTPTK             SB3    B0                                                     OUTPTK     1002    A
OUTPTK             SPACE  1                                                     OUTPTK     1003    A
OUTPTK   RETURN    BSS    1              ENTRY/EXIT INFORMATION FOR INVOKED     OUTPTK     1004    A
OUTPTK                                      PROCESSORS.                         OUTPTK     1005    A
OUTPTK             SPACE  1                                                     OUTPTK     1006    A
OUTPTK             JP     B7             JUMP TO THE APPROPRIATE PROCESSOR.     OUTPTK     1007    A
OUTPTK    ILLEGAL  SPACE  3                                                     OUTPTK     1008    A
OUTPTK   **        ILLEGAL - PROCESS ILLEGAL FORMAT SPECIFICATION               OUTPTK     1009    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OUTPTK

```
OUTPTK                                                                  OUTPTK    1010    A
OUTPTK    ILLEGAL  INVOKE SLASH                                         OUTPTK    1011    A
OUTPTK             EQ     *+1S17      ABORT                             OUTPTK    1012    A
OUTPTK             TITLE                  UTILITY: D E C I M A L        OUTPTK    1013    A
OUTPTK   ************************************************************* OUTPTK    1014    A
OUTPTK   *                                                         *   OUTPTK    1015    A
OUTPTK   *                       DECIMAL                           *   OUTPTK    1016    A
OUTPTK   *                                                         *   OUTPTK    1017    A
OUTPTK   *               THIS FUNCTION WILL CONVERT A STRING OF DECIMAL DIGITS *   OUTPTK    1018    A
OUTPTK   *             (FROM THE FORMAT) TO BINARY.  CALLING SEQUENCE --   *   OUTPTK    1019    A
OUTPTK   *                                                         *   OUTPTK    1020    A
OUTPTK   *                               RJ  DECIMAL               *   OUTPTK    1021    A
OUTPTK   *                                                         *   OUTPTK    1022    A
OUTPTK   *             WHERE THE FOLLOWING REGISTERS HAVE BEEN SET:    *   OUTPTK    1023    A
OUTPTK   *                               X5 -- 'DIGMASK'           *   OUTPTK    1024    A
OUTPTK   *                               X6 -- INITIAL VALUE OF NUMBER *   OUTPTK    1025    A
OUTPTK   *                               X7 -- 7777 7777 7777 7777 7700B *   OUTPTK    1026    A
OUTPTK   *                                                         *   OUTPTK    1027    A
OUTPTK   *             UPON RETURN REGISTER  X6  WILL CONTAIN THE CONVERTED *   OUTPTK    1028    A
OUTPTK   *             NUMBER AND REGISTER  X4  WILL CONTAIN THE TERMINATING *   OUTPTK    1029    A
OUTPTK   *             (NON-DIGIT) CHARACTER, THE FORMAT POINTERS WILL NOT *   OUTPTK    1030    A
OUTPTK   *             HAVE BEEN UPDATED TO REFLECT THE EXTRACTION OF THE *   OUTPTK    1031    A
OUTPTK   *             TERMINATING CHARACTER.                      *   OUTPTK    1032    A
OUTPTK   *                                                         *   OUTPTK    1033    A
OUTPTK   *********************************************************** OUTPTK    1034    A
OUTPTK             SPACE  1                                            OUTPTK    1035    A
OUTPTK   DEC.1     IX3    X6+X6          MULTIPLY THE ACCUMULATED NUMBER BY TEN OUTPTK    1036    A
OUTPTK             LX6    3                AND ADD THE DIGIT JUST ENCOUNTERED.  OUTPTK    1037    A
OUTPTK             SX4    X4-1R0                                      OUTPTK    1038    A
OUTPTK             IX3    X6+X3                                       OUTPTK    1039    A
OUTPTK             SB2    B2-6                                        OUTPTK    1040    A
OUTPTK             IX6    X3+X4                                       OUTPTK    1041    A
OUTPTK             PL     B2,DEC.2                                    OUTPTK    1042    A
OUTPTK             SB2    B6+B2                                       OUTPTK    1043    A
OUTPTK             SA2    A2+B1                                       OUTPTK    1044    A
OUTPTK   DEC.2     AX3    X2,B2          EXTRACT THE NEXT FORMAT CHARACTER AND OUTPTK    1045    A
OUTPTK             BX4    -X7*X3         IF IT IS A DIGIT THE CYCLE.  OUTPTK    1046    A
OUTPTK             SB5    X4                                          OUTPTK    1047    A
OUTPTK             LX3    X5,B5                                       OUTPTK    1048    A
OUTPTK             MI     X3,DEC.1                                    OUTPTK    1049    A
OUTPTK                                      OTHERWISE RETURN.         OUTPTK    1050    A
OUTPTK             SPACE  1                                           OUTPTK    1051    A
OUTPTK   DECIMAL   BSS    1                                           OUTPTK    1052    A
OUTPTK             AX3    X2,B2                                       OUTPTK    1053    A
OUTPTK             BX4    -X7*X3                                      OUTPTK    1054    A
OUTPTK             SB5    X4                                          OUTPTK    1055    A
OUTPTK             LX3    X5,B5                                       OUTPTK    1056    A
OUTPTK             MI     X3,DEC.1                                    OUTPTK    1057    A
OUTPTK             ZR     DECIMAL                                     OUTPTK    1058    A
OUTPTK             TITLE                  UTILITY:  S P A C E         OUTPTK    1059    A
OUTPTK   ********************************************************** OUTPTK    1060    A
OUTPTK   *                                                        *  OUTPTK    1061    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OUTPTK

```
OUTPTK   *                    S P A C E                                  *        OUTPTK     1062     A
OUTPTK   *                                                               *        OUTPTK     1063     A
OUTPTK   *               THIS ROUTINE WILL APPEND A DESIGNATED NUMBER OF *        OUTPTK     1064     A
OUTPTK   *             SPACES TO THE CURRENT OUTPUT LINE AND RETURN WITH THE *    OUTPTK     1065     A
OUTPTK   *             LINE POINTERS UPDATED.                            *        OUTPTK     1066     A
OUTPTK   *                                                               *        OUTPTK     1067     A
OUTPTK   *               CALLING SEQUENCE--                              *        OUTPTK     1068     A
OUTPTK   *                                                               *        OUTPTK     1069     A
OUTPTK   *                    CALL  SPACE                                *        OUTPTK     1070     A
OUTPTK   *                                                               *        OUTPTK     1071     A
OUTPTK   *             WHERE REGISTER X6 HAS BEEN SET TO THE NUMBER OF BITS TO *  OUTPTK     1072     A
OUTPTK   *             BE FILLED BY BLANK CHARACTERS (I.E. SIX TIMES THE NUMBER *  OUTPTK     1073     A
OUTPTK   *             OF BLANK CHARACTERS TO BE APPENDED).              *        OUTPTK     1074     A
OUTPTK   *                                                               *        OUTPTK     1075     A
OUTPTK   *               REGISTERS USED--  X4, X5, X6, X7                *        OUTPTK     1076     A
OUTPTK   *                                 A4, A5, A6                    *        OUTPTK     1077     A
OUTPTK   *                                     B5,     B7                *        OUTPTK     1078     A
OUTPTK   *                                                               *        OUTPTK     1079     A
OUTPTK   *             THE NORMAL 'OUTPTK' REGISTER USAGES ARE ASSUMED   *        OUTPTK     1080     A
OUTPTK   *             (I.E. B1=1, B6=60, X0=CURRENT OUTPUT WORD ADDRESS, ETC.).*  OUTPTK     1081     A
OUTPTK   *                                                               *        OUTPTK     1082     A
OUTPTK   *****************************************************************        OUTPTK     1083     A
OUTPTK            SPACE  1                                                       OUTPTK     1084     A
OUTPTK   SPACE    ROUTINE                                                        OUTPTK     1085     A
OUTPTK            SB7    X6              (SET REGISTER B7 TO THE SPACE COUNT AND OUTPTK     1086     A
OUTPTK            SB5    B6-B3             REGISTER X5 TO AN ALL BLANK WORD.)   OUTPTK     1087     A
OUTPTK            SA5    =10H                                                   OUTPTK     1088     A
OUTPTK            LT     B7,B5,SPACE.3   IF THE NUMBER OF COLUMNS TO SPACE LIES OUTPTK     1089     A
OUTPTK                                     WITHIN THE CURRENT OUTPUT WORD BUT   OUTPTK     1090     A
OUTPTK                                     DOES NOT FILL IT, GO TO 'SPACE.3',   OUTPTK     1091     A
OUTPTK            MX4    1               OTHERWISE FILL OUT THE CURRENT OUTPUT  OUTPTK     1092     A
OUTPTK            SB7    B7-B5             WORD WITH BLANKS.                    OUTPTK     1093     A
OUTPTK            SB5    B3-B1                                                  OUTPTK     1094     A
OUTPTK            AX4    X4,B5                                                  OUTPTK     1095     A
OUTPTK            BX7    X4                                                     OUTPTK     1096     A
OUTPTK            AX4    59                                                     OUTPTK     1097     A
OUTPTK            BX7    X4*X7                                                  OUTPTK     1098     A
OUTPTK            BX6    -X7*X5                                                 OUTPTK     1099     A
OUTPTK            SA4    X0                                                     OUTPTK     1100     A
OUTPTK            BX7    X7*X4                                                  OUTPTK     1101     A
OUTPTK            IX6    X7+X6                                                  OUTPTK     1102     A
OUTPTK            SA6    X0                                                     OUTPTK     1103     A
OUTPTK            LT     B7,B6,SPACE.2   IF THE NUMBER OF BLANKS LEFT TO BE     OUTPTK     1104     A
OUTPTK                                     APPENDED WILL NOT FILL A COMPLETE    OUTPTK     1105     A
OUTPTK                                     WORD THEN GO TO 'SPACE.2',           OUTPTK     1106     A
OUTPTK            SPACE  1                                                       OUTPTK     1107     A
OUTPTK            BX6    X5              OTHERWISE FILL AS MANY COMPLETE WORDS  OUTPTK     1108     A
OUTPTK   SPACE.1  SB7    B7-B6             AS NECESSARY UNTIL ONLY A PARTIAL    OUTPTK     1109     A
OUTPTK            SA6    A6+B1             (OR NO) WORD REMAINS.                OUTPTK     1110     A
OUTPTK            GE     B7,B6,SPACE.1                                          OUTPTK     1111     A
OUTPTK            SPACE  1                                                       OUTPTK     1112     A
OUTPTK   SPACE.2  SX0    A6+B1           UPDATE THE OUTPUT LINE POINTERS.       OUTPTK     1113     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OUTPTK

```
OUTPTK            SB3    B0                                                    OUTPTK      1114     A
OUTPTK            SPACE  1                                                     OUTPTK      1115     A
OUTPTK   SPACE.3  ZR     B7,SPACE        RETURN IF THERE ARE NO MORE SPACES TO OUTPTK      1116     A
OUTPTK                                     APPEND,                             OUTPTK      1117     A
OUTPTK            SB5    B6-B3           OTHERWISE APPEND THE SPACES, UPDATE THE OUTPTK    1118     A
OUTPTK            MX4    1                 LINE POINTERS AND RETURN.             OUTPTK      1119     A
OUTPTK            SB3    B3+B7                                                 OUTPTK      1120     A
OUTPTK            SB7    B7-B1                                                 OUTPTK      1121     A
OUTPTK            AX7    X4,B7                                                 OUTPTK      1122     A
OUTPTK            SA4    X0                                                    OUTPTK      1123     A
OUTPTK            LX7    X7,B5                                                 OUTPTK      1124     A
OUTPTK            BX5    X7*X5                                                 OUTPTK      1125     A
OUTPTK            BX4    -X7*X4                                                OUTPTK      1126     A
OUTPTK            IX6    X4+X5                                                 OUTPTK      1127     A
OUTPTK            SA6    X0                                                    OUTPTK      1128     A
OUTPTK            ZR     SPACE                                                 OUTPTK      1129     A
OUTPTK    SETST   SPACE  3,14                                                  OUTPTK      1130     A
OUTPTK   **       SETST - SET SUB TITLE LINE                                   OUTPTK      1131     A
OUTPTK   *                                                                     OUTPTK      1132     A
OUTPTK   *        CALL SETST( WDS , "SUBTITLE" )                               OUTPTK      1133     A
OUTPTK                                                                         OUTPTK      1134     A
OUTPTK    SETST0  SX6    1                                                     OUTPTK      1135     A
OUTPTK            SA6    =XN$STITL                                             OUTPTK      1136     A
OUTPTK            SX7    2R0                                                   OUTPTK      1137     A
OUTPTK            LX7    48                                                    OUTPTK      1138     A
OUTPTK            SA7    =XF$STITL                                             OUTPTK      1139     A
OUTPTK                                                                         OUTPTK      1140     A
OUTPTK    SETST   ENTRY. **                                                    OUTPTK      1141     A
OUTPTK            SA4    X1                                                    OUTPTK      1142     A
OUTPTK            SB1    1                                                     OUTPTK      1143     A
OUTPTK            ZR     X4,SETST0   IF WDS = 0    */ NO SUBSTITLE             OUTPTK      1144     A
OUTPTK            BX6    X4                                                    OUTPTK      1145     A
OUTPTK            SA6    N$STITL                                               OUTPTK      1146     A
OUTPTK            SA2    A1+B1                                                 OUTPTK      1147     A
OUTPTK            MOVE   X4,X2,F$STITL                                         OUTPTK      1148     A
OUTPTK            SA1    =XN$LINES   LINES WRITTEN                             OUTPTK      1149     A
OUTPTK            SA3    =XCP.PS                                               OUTPTK      1150          I
         -CCGA012
OUTPTK            SA3    =XCO.PS                                               CCGA012        4     A
OUTPTK            IX5    X3-X1                                                 OUTPTK      1151     A
OUTPTK            SX1    X5-5                                                  OUTPTK      1152     A
OUTPTK            PL     X1,SETST1   IF MORE THAN 4 LEFT                       OUTPTK      1153     A
OUTPTK            SX6    X3+1        FORCE AN EJECT                            OUTPTK      1154     A
OUTPTK            SA6    A1                                                    OUTPTK      1155     A
OUTPTK                                                                         OUTPTK      1156     A
OUTPTK    SETST1  LISTL  (=2C  ),1                                             OUTPTK      1157     A
OUTPTK            SA1    N$STITL                                               OUTPTK      1158     A
OUTPTK            LISTL  F$STITL,X1                                            OUTPTK      1159     A
OUTPTK            EQ     SETST                                                 OUTPTK      1160     A
OUTPTK    REMARK  SPACE  3,8                                                   OUTPTK      1161     A
OUTPTK    REMARK  ENTRY.                                                       OUTPTK      1162     A
OUTPTK            MESSAGE  X1                                                  OUTPTK      1163     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN OUTPTK

```
OUTPTK            EQ     REMARK                                                OUTPTK     1164     A
OUTPTK                                                                         OUTPTK     1165     A
OUTPTK    .TEST   ENDIF                                                        OUTPTK     1166     A
OUTPTK                                                                         OUTPTK     1167     A
OUTPTK    GOTOER. ENTRY.                                                       OUTPTK     1168     A
OUTPTK            EQ     *+1S17                                                OUTPTK     1169     A
OUTPTK    SPA.    SPACE  3,14                                                  OUTPTK     1170     A
OUTPTK   **       SPA. - SUBSTITUTE PARAMETER ADDRESS"S                        OUTPTK     1171     A
OUTPTK                                                                         OUTPTK     1172     A
OUTPTK    SPA.    ENTRY.                                                       OUTPTK     1173     A
OUTPTK   +        UX4    B3,X3                                                 OUTPTK     1174     A
OUTPTK            SB5    A0                                                    OUTPTK     1175     A
OUTPTK            AX4    42                                                    OUTPTK     1176     A
OUTPTK            SA1    X4+B5                                                 OUTPTK     1177     A
OUTPTK                                                                         OUTPTK     1178     A
OUTPTK    SPA1    SA2    X3                                                    OUTPTK     1179     A
OUTPTK            UX4    B3,X3                                                 OUTPTK     1180     A
OUTPTK            MX0    42                                                    OUTPTK     1181     A
OUTPTK            SB4    X1                                                    OUTPTK     1182     A
OUTPTK                                                                         OUTPTK     1183     A
OUTPTK    SPA2    AX4    18                                                    OUTPTK     1184     A
OUTPTK            SA3    A3+1                                                  OUTPTK     1185     A
OUTPTK            SX5    X4+B4                                                 OUTPTK     1186     A
OUTPTK            LX1    B3,X0                                                 OUTPTK     1187     A
OUTPTK            BX6    -X0*X5                                                OUTPTK     1188     A
OUTPTK            SB2    A2                                                    OUTPTK     1189     A
OUTPTK            LX5    B3,X6                                                 OUTPTK     1190     A
OUTPTK            BX4    X1*X2                                                 OUTPTK     1191     A
OUTPTK            SA2    X3                                                    OUTPTK     1192     A
OUTPTK            BX6    X4+X5                                                 OUTPTK     1193     A
OUTPTK            UX4    B3,X3                                                 OUTPTK     1194     A
OUTPTK            SA6    B2+                                                   OUTPTK     1195     A
OUTPTK            MI     X3,SPA2                                               OUTPTK     1196     A
OUTPTK                                                                         OUTPTK     1197     A
OUTPTK            AX4    42                                                    OUTPTK     1198     A
OUTPTK            SA1    X4+B5                                                 OUTPTK     1199     A
OUTPTK            NZ     B3,SPA1                                               OUTPTK     1200     A
OUTPTK            EQ     SPA.                                                  OUTPTK     1201     A
OUTPTK                                                                         OUTPTK     1202     A
OUTPTK            END                                                          OUTPTK     1203     A
OUTPTK   *ENDIF                                                                OUTPTK     1204     A
```

SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - OUTPTK

IDENTIFIER	TOTAL	ACTIVE
OUTPTK	1204	1202
CCG27	1	1
CCGA012	1	1

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN *CT*

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN *CT*

*CT*      *DECK     *CT*                                                    *CT*          1     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - *CT*

          IDENTIFIER                TOTAL    ACTIVE

          *CT*                          1         1

          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFTBLD

CFTBLD    *COMDECK  CFTBLD                                                  CFTBLD        1     A
CFTBLD     CFTBLD   CTEXT  CFTBLD - CONTROL FLOW TABLE DEFINITIONS          CFTBLD        2     A
CFTBLD                                                                      CFTBLD        3     A
CFTBLD    **        CF. - CONTROL FLOW TABLE FORMAT                         CFTBLD        4          I
          -CCG16
CFTBLD                                                                      CFTBLD        5          I
          -CCG16
CFTBLD              DESCRIBE CF.,60                                         CFTBLD        6          I
          -CCG16
CFTBLD     JP       DEFINE 1                                                CFTBLD        7          I
          -CCG16
CFTBLD              DEFINE 11                                               CFTBLD        8          I
          -CCG16
CFTBLD     FROM     DEFINE 18          BLOCK NUMBER OF PRED                 CFTBLD        9          I
          -CCG16
CFTBLD              DEFINE 12                                               CFTBLD       10          I
          -CCG16
CFTBLD     TO       DEFINE 18          BLOCK NUMBER OF SUCC                 CFTBLD       11          I
          -CCG16
CFTBLD     GST      SPACE  3                                                CFTBLD       12          I
          -CCG16
CFTBLD    **        THE GRAPH STRUCTURE TABLES CONSIST OF ( FOR EACH GRAPH ) -  CFTBLD  13     A
CFTBLD    *         GA, GB WORDS, THE EDGE INDEX TABLE, THE EDGE TABLE, AND     CFTBLD  14     A
CFTBLD    *         INTERVAL LISTS, WHICH ARE TERMINATED BY A ZERO WORD.        CFTBLD  15     A
CFTBLD                                                                      CFTBLD       16     A
CFTBLD    **        GRAPH HEADER WORDS                                      CFTBLD       17     A
CFTBLD                                                                      CFTBLD       18     A
CFTBLD              DESCRIBE GA.,60                                         CFTBLD       19     A
CFTBLD              DEFINE 24                                               CFTBLD       20     A
CFTBLD     LB       DEFINE 18          LINK BACKWARDS ( LENGTH OF LAST GRAPH )  CFTBLD  21     A
CFTBLD     LF       DEFINE 18          LINK FOWARD ( LENGTH OF THIS GRAPH )     CFTBLD  22     A
CFTBLD                                                                      CFTBLD       23     A
CFTBLD              DESCRIBE GB.,60                                         CFTBLD       24     A
CFTBLD              DEFINE 6                                                CFTBLD       25     A
CFTBLD     NLP      DEFINE 18          N.LOOPS                              CFTBLD       26     A
CFTBLD     NB       DEFINE 18          N.BLOCKS IN GRAPH                    CFTBLD       27     A
CFTBLD     ILI      DEFINE 18          INTERVAL LIST INDEX ( REL TO *EII* )  CFTBLD  28     A
CFTBLD     EIT      SPACE  2,10                                             CFTBLD       29     A
CFTBLD    **        EI. - EDGE INDEX TABLE                                  CFTBLD       30     A
CFTBLD                                                                      CFTBLD       31     A
CFTBLD              DESCRIBE EI.,60                                         CFTBLD       32     A
CFTBLD     PRI      DEFINE 18          PREDECESSOR INDEX  ( REL TO EIB )    CFTBLD       33     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFTBLD

```
CFTBLD    NP       DEFINE 12          N.PRED ( PACKED )                          CFTBLD        34     A
CFTBLD    SUI      DEFINE 18          SUCCESSOR INDEX                            CFTBLD        35     A
CFTBLD    NS       DEFINE 12          N.SUCC ( PACKED )                          CFTBLD        36     A
CFTBLD    EIT      SPACE  2,10                                                   CFTBLD        37     A
CFTBLD   **        ET. - EDGE TABLE FORMAT                                       CFTBLD        38     A
CFTBLD                                                                           CFTBLD        39     A
CFTBLD             DESCRIBE ET.,60                                               CFTBLD        40     A
CFTBLD    JP       DEFINE 1           JUMP EDGE ( SUCCESSOR EDGES )              CFTBLD        41     A
CFTBLD             DEFINE 11                                                     CFTBLD        42     A
CFTBLD    PRED     DEFINE 18          BLOCK NUMBER OF PRED                       CFTBLD        43     A
CFTBLD             DEFINE 12                                                     CFTBLD        44     A
CFTBLD    SUCC     DEFINE 18          BLOCK NUMBER OF SUCCESSOR                  CFTBLD        45     A
CFTBLD    IP       SPACE  3,14                                                   CFTBLD        46     A
CFTBLD   **        IP. - INTERVAL PROPERTIES                                     CFTBLD        47     A
CFTBLD                                                                           CFTBLD        48     A
CFTBLD             DESCRIBE IP.,60                                               CFTBLD        49     A
CFTBLD    GL       DEFINE 1           SET IF REGION NEEDS A *GL* WHEN OPTIMIZED  CFTBLD        50     A
CFTBLD             DEFINE 5                                                      CFTBLD        51          I
         -FCC0408
CFTBLD    FT       DEFINE 1           FALL THROUGH TO HEADER NODE                FCC0408        7     A
CFTBLD             DEFINE 4                                                      FCC0408        8     A
CFTBLD    LB       DEFINE 18          BACKWARDS LINK ( NODES+1 IN PREV INTERVAL ) CFTBLD       52          I
         -FCC0392
CFTBLD    NIP      DEFINE 18          N. INTERNAL PRED OF HEADER NODE            FCC0392       11     A
CFTBLD    PB       DEFINE 18          BLOCK NUMBER OF PROLOGUE BLOCK ( *SCR* " 0 CFTBLD        53     A
CFTBLD    NODE     DEFINE 18          NUMBER OF NODES IN INTERVAL                CFTBLD        54     A
CFTBLD    IM       SPACE  2                                                      CFTBLD        55     A
CFTBLD   **        IM. - INTERVAL MEMBER PROPERTIES                              CFTBLD        56     A
CFTBLD                                                                           CFTBLD        57     A
CFTBLD             DESCRIBE IM.,60                                               CFTBLD        58     A
CFTBLD    SCR      DEFINE 1           IN STRONGLY CONNECTED REGION               CFTBLD        59     A
CFTBLD    AP       DEFINE 1           ARTICULATION POINT                         CFTBLD        60     A
CFTBLD    LN       DEFINE 1           LATCHING NODE ( JUMPS BACK TO LOOP HEADER ) CFTBLD       61     A
CFTBLD    DOM      DEFINE 1           NODE IS A DOMINATOR                        CFTBLD        62     A
CFTBLD    EN       DEFINE 1           INTERVAL EXIT NODE                         CFTBLD        63     A
CFTBLD    HN       DEFINE 1           HEADER NODE                                CFTBLD        64     A
CFTBLD    BI       DEFINE 18          INDEX TO BLOCK INFO IN *BIT*               CFTBLD        65     A
CFTBLD    IBD      DEFINE 18          IMMEDIATE BACK DOMINATOR ( REL TO *HN* )   CFTBLD        66     A
CFTBLD    BN       DEFINE 18          NODE NUMBER OF BLOCK                       CFTBLD        67     A
CFTBLD                                                                           CFTBLD        68     A
CFTBLD    APLN     DEQU   LN,2        AP AND LN BITS                             CFTBLD        69     A
CFTBLD                                                                           CFTBLD        70     A
CFTBLD             ENDX                                                          CFTBLD        71     A
```

SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - CFTBLD

IDENTIFIER	TOTAL	ACTIVE
CFTBLD	71	60
FCC0392	1	1
FCC0408	2	2

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFTBLD

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN IOTBLD

```
IOTBLD   *COMDECK  IOTBLD                                                IOTBLD        1     A
IOTBLD    IOTBLD   CTEXT  IOTBLD - RANDOM I/O TABLE FORMAT DEFINITIONS   IOTBLD        2     A
IOTBLD                                                                   IOTBLD        3     A
IOTBLD   **        RI. - RANDOM INDEX WORD                               IOTBLD        4     A
IOTBLD                                                                   IOTBLD        5     A
IOTBLD             DESCRIBE RI.,60    RIW(FLAGS,LEN,[FWA],[LCM],[OFS,RA])  IOTBLD      6     A
IOTBLD    IC       DEFINE 1           =1  IN CORE, =0 ON MS             IOTBLD        7     A
IOTBLD    AT       DEFINE 1           ADDRESS TYPE, =0 DISK, =1 LCM     IOTBLD        8     A
IOTBLD             DEFINE 1                                             IOTBLD        9     A
IOTBLD    LEN      DEFINE 18          BLOCK LENGTH (INCORE COPY)        IOTBLD       10     A
IOTBLD    OFS      DEFINE 9           BLOCK OFFSET IN RECORD            IOTBLD       11     A
IOTBLD    RA       DEFINE 30          DISK ADDRESS OF BLOCK             IOTBLD       12     A
IOTBLD                                                                   IOTBLD       13     A
IOTBLD             DESCRIBE RI.,36                                      IOTBLD       14     A
IOTBLD    FWA      DEFINE 18          FWA OF BLOCK IN *BLK*             IOTBLD       15     A
IOTBLD    LCM      DEFINE 18          LCM ADDRESS OF BLOCK              IOTBLD       16     A
IOTBLD    BAW      SPACE  2,8                                           IOTBLD       17     A
IOTBLD   **        BA. - BLOCK ADDRESS INFORMATION WORD                 IOTBLD       18     A
IOTBLD                                                                   IOTBLD       19     A
IOTBLD             DESCRIBE BA.,60    BAW(FLAGS,BI,LEN,FWA)             IOTBLD       20     A
IOTBLD    BLK      DEFINE 1           =1 IF BLOCK IS IN *BLK*           IOTBLD       21     A
IOTBLD    PRI      DEFINE 1           BLOCK PRIORITY FOR PURGING        IOTBLD       22     A
IOTBLD    HB       DEFINE 1           HOLDING BLOCK                     IOTBLD       23     A
IOTBLD    FW       DEFINE 1           FINAL WRITE FLAG                  IOTBLD       24     A
IOTBLD             DEFINE 2                                             IOTBLD       25     A
IOTBLD    BI       DEFINE 18          *BIT* INDEX ( = BN*2 )            IOTBLD       26     A
IOTBLD    LEN      DEFINE 18          BLOCK LENGTH  (INCORE COPY)       IOTBLD       27     A
IOTBLD    FWA      DEFINE 18          FWA OF BLOCK IN *BLK*             IOTBLD       28     A
IOTBLD                                                                   IOTBLD       29     A
IOTBLD    HBB      DEQU   HB,2        HB AND PRI BITS                   IOTBLD       30     A
IOTBLD    SAV      DEQU   BI,23                                         IOTBLD       31     A
IOTBLD    SVL      DEQU   LEN,41                                        IOTBLD       32     A
IOTBLD    BHW      SPACE  2,8                                           IOTBLD       33     A
IOTBLD   **        BH. - BLOCK HEADER INFORMATION WORD                  IOTBLD       34     A
IOTBLD                                                                   IOTBLD       35     A
IOTBLD             DESCRIBE BH.,60    BHW(FLAGS,BI,BST,LEN)             IOTBLD       36     A
IOTBLD    AV       DEFINE 1           =1 BLOCK IS AVAILABLE SPACE       IOTBLD       37     A
IOTBLD    PRI      DEFINE 1           BLOCK PRIORITY FOR PURGING        IOTBLD       38     A
IOTBLD    HB       DEFINE 1           HOLDING BLOCK                     IOTBLD       39     A
IOTBLD    FW       DEFINE 1           FINAL WRITE FLAG                  IOTBLD       40     A
IOTBLD             DEFINE 2                                             IOTBLD       41     A
IOTBLD    BI       DEFINE 18          BIT INDEX, (BN*2)                 IOTBLD       42     A
IOTBLD    BST      DEFINE 18          INDEX TO BST TABLE                IOTBLD       43     A
IOTBLD    LEN      DEFINE 18          BLOCK LENGTH                      IOTBLD       44     A
IOTBLD                                                                   IOTBLD       45     A
IOTBLD   **        BT. - BLOCK TRAILER WORD                             IOTBLD       46     A
```

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
            LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN IOTBLD

IOTBLD                                                                  IOTBLD        47     A
IOTBLD            DESCRIBE BT.,60    BTW(FLAGS,LKB,LKF,LEN)             IOTBLD        48     A
IOTBLD    AV      DEFINE 1           =1 BLOCK IS AVAILABLE SPACE        IOTBLD        49     A
IOTBLD            DEFINE 5                                              IOTBLD        50     A
IOTBLD    LKB     DEFINE 18          ABS LINK BACKWARD                  IOTBLD        51     A
IOTBLD    LKF     DEFINE 18          ABS LINK FORWARD                   IOTBLD        52     A
IOTBLD    LEN     DEFINE 18          BLOCK LENGTH (INCLUDING EXTRA WORDS) IOTBLD      53     A
IOTBLD                                                                  IOTBLD        54     A
IOTBLD            ENDX                                                  IOTBLD        55     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - IOTBLD

            IDENTIFIER              TOTAL    ACTIVE

            IOTBLD                     55        55

            LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPOCOM

GPOCOM    *COMDECK  GPOCOM                                              GPOCOM         1     A
GPOCOM     CLCM     TITLE  CLCM - COROUTINE LOOP CONTROL MACRO         GPOCOM         2     A
GPOCOM    **        CLCM - COROUTINE LOOP CONTROL MACRO                GPOCOM         3     A
GPOCOM    *                                                            GPOCOM         4     A
GPOCOM    *         CLCM   ROUTINE,(BSW-FLAGS)                         GPOCOM         5     A
GPOCOM    *                                                            GPOCOM         6     A
GPOCOM    *         CALL *ROUTINE* FROM *PCC* IF *BSW-FLAGS* " 0 , FOR EACH BLOCK GPOCOM 7  A
GPOCOM    *         IN THE REGION ( *SCR* ) BEING PROCESSED.           GPOCOM         8     A
GPOCOM    *         IF *FIELDS* IS ABSENT, THEN CALL *ROUTINE* UNCONDITIONALLY. GPOCOM 9  A
GPOCOM                                                                 GPOCOM        10     A
GPOCOM     CLCM     MACRO  RTN,FIELDS                                  GPOCOM        11     A
GPOCOM              LOCAL  CCW                                         GPOCOM        12     A
GPOCOM              SX7    =X_RTN                                      GPOCOM        13     A
GPOCOM                                                                 GPOCOM        14     A
GPOCOM              IFC    EQ,/FIELDS//,2                              GPOCOM        15     A
GPOCOM              SX5    0                                           GPOCOM        16     A
GPOCOM              ELSE   4                                           GPOCOM        17     A
GPOCOM              SA5    CCW                                         GPOCOM        18     A
GPOCOM              USE    CLCM                                        GPOCOM        19     A
GPOCOM     CCW      BFMW   BS,(FIELDS)                                 GPOCOM        20     A
GPOCOM              USE    *                                           GPOCOM        21     A
GPOCOM                                                                 GPOCOM        22     A
GPOCOM              RJ     =XPCC                                       GPOCOM        23           I
            -CCGNC
GPOCOM              RJ     =XPCC#                                      CCGNC        480     A
GPOCOM              ENDM                                               GPOCOM        24     A
GPOCOM     FBVA     SPACE  2                                           GPOCOM        25     A
GPOCOM    **        FBVA - FORM BLOCK BIT VECTOR ADDRESS               GPOCOM        26     A
GPOCOM    *                                                            GPOCOM        27     A
GPOCOM    *         ENTRY  (BN) = REGISTER WITH *BI* INDEX OF BLOCK    GPOCOM        28     A
GPOCOM    *                                                            GPOCOM        29     A
GPOCOM    *         EXIT   (X7) = BVA = O.BVT + BVI[BIT(BN)]           GPOCOM        30     A
GPOCOM    *                (X4) = BIT(BN)                              GPOCOM        31     A
GPOCOM    *                (B2) = VL = BVL + 1                         GPOCOM        32     A

            0         1         2         3         4         5         6         7         8
            12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPOCOM

```
GPOCOM                                                                    GPOCOM        33     A
GPOCOM    FBVA     MACRO  BN                                              GPOCOM        34     A
GPOCOM             R=     B3,BN                                           GPOCOM        35     A
GPOCOM             SA1    O.BVT                                           GPOCOM        36     A
GPOCOM             SA2    BVL                                             GPOCOM        37     A
GPOCOM             SA3    O.BIT                                           GPOCOM        38     A
GPOCOM             SB2    X2+B1       VL = BVL + 1                        GPOCOM        39     A
GPOCOM             SA4    X3+B3       BIT(BN)                             GPOCOM        40     A
GPOCOM             LX4    -BI.BVIP                                        GPOCOM        41     A
GPOCOM             IX7    X1+X4       BVA = O.BVT + BVI[BIT(BN)]          GPOCOM        42     A
GPOCOM             LX4    BI.BVIP                                         GPOCOM        43     A
GPOCOM             ENDM                                                   GPOCOM        44     A
GPOCOM    PRNTABV  SPACE  2                                               GPOCOM        45     A
GPOCOM   **        PRNTABV - PRINT A BIT VECTOR                           GPOCOM        46     A
GPOCOM   *                                                                GPOCOM        47     A
GPOCOM   *         PRNTABV  LAB,(LIST OF VECTOR ADDRESS"S)                GPOCOM        48     A
GPOCOM                                                                    GPOCOM        49     A
GPOCOM    PRNTABV  MACRO  LAB,BVA                                         GPOCOM        50     A
GPOCOM             IF     DEF,/DEBUG/LAB,3                                GPOCOM        51     A
GPOCOM             IRP    BVA                                             GPOCOM        52     A
GPOCOM             DCALL  PRNTABV,(=5L BVA ,[_BVA_])                      GPOCOM        53     A
GPOCOM             IRP                                                    GPOCOM        54     A
GPOCOM             ENDM                                                   GPOCOM        55     A
GPOCOM    MICROS   SPACE  3                                               GPOCOM        56     A
GPOCOM   *         SOME POPULAR REGISTER ASSIGNMENTS                      GPOCOM        57     A
GPOCOM                                                                    GPOCOM        58     A
GPOCOM    TB       MICRO  1,,/B7/      O.SEQ + 3                         GPOCOM        59     A
GPOCOM    RN       MICRO  1,,/-X0*/    R-NUMBER EXTRACTOR MASK           GPOCOM        60     A
GPOCOM    EXIT     SPACE  3,14                                            GPOCOM        61     A
GPOCOM   *         E. - NODE NUMBER TO *BIT* INDEX TRANSITION TABLE FORMAT GPOCOM        62     A
GPOCOM                                                                    GPOCOM        63     A
GPOCOM             DESCRIBE E.,60                                         GPOCOM        64     A
GPOCOM    SCR      DEFINE 1            IN *SCR*                           GPOCOM        65     A
GPOCOM    JP       DEFINE 1            SUCC REACHED BY A *JP* EDGE        GPOCOM        66     A
GPOCOM    ALLP     DEFINE 1            ALL PRED OF EXIT NODE IN *SCR*     GPOCOM        67     A
GPOCOM    NPRED    DEFINE 9            N.PRED IN SCR OF SUCCESSOR NODE    GPOCOM        68     A
GPOCOM    PRED     DEFINE 15           NODE NUM OF FIRST PRED IN SCR TO REACH SUCC GPOCOM 69           I
         -FCC0985
GPOCOM    PRED     DEFINE 15           *BI* OF FIRST PRED IN SCR TO REACH SUCC FCC0985    5     A
GPOCOM    LINK     DEFINE 18           LINK TO NEXT ON SUCC LIST          GPOCOM        70     A
GPOCOM    BI       DEFINE 15           *BIT* INDEX OF NODE NUMBER I       GPOCOM        71     A
GPOCOM    PS       SPACE  3,14                                            GPOCOM        72     A
GPOCOM   **        PS. - POST STORE INFO FORMAT ( *PSI* )                 GPOCOM        73     A
GPOCOM   *                                                                GPOCOM        74     A
GPOCOM   *         FIRST WORD OF A *PSI* ENTRY IS THE NUMBER OF POST STORES. GPOCOM        75     A
GPOCOM   *         THE REMAINNG WORDS SPECIFY THE VARIABLES TO BE STORED OUT AND GPOCOM    76     A
GPOCOM   *         THE REGISTERS THAT THEY ARE IN.                        GPOCOM        77     A
GPOCOM                                                                    GPOCOM        78     A
GPOCOM             DESCRIBE PS.,60                                        GPOCOM        79     A
GPOCOM    REG      DEFINE 6            REGISTER NUMBER THAT VAR IS IN ( *SO* ORDER GPOCOM 80     A
GPOCOM    UDI      DEFINE 18           UDI INDEX OF VARIABLE              GPOCOM        81     A
GPOCOM    CAIH     DEFINE 36           BASE/BIAS OF SYMBOL                GPOCOM        82     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPOCOM

```
GPOCOM    IIT      EJECT                                                    GPOCOM        83     A
GPOCOM   **        IIT - INCREMENT INFO TABLE                               GPOCOM        84     A
GPOCOM   *                                                                  GPOCOM        85     A
GPOCOM   *         *IIT* CONSISTS OF 3 LISTS -                              GPOCOM        86     A
GPOCOM   *                                                                  GPOCOM        87     A
GPOCOM   *         A) THE LIST OF *RD* INCREMENT VALUES                     GPOCOM        88     A
GPOCOM   *                HBP OF THE LINK WORD OF THE *RD* POINTS TO *IIT*  GPOCOM        89     A
GPOCOM   *         (IIT) = 12/P(TYPE),48/VAL                                GPOCOM        90     A
GPOCOM   *                TYPE        VALUE                                 GPOCOM        91     A
GPOCOM   *                0           S    RI,VAL                           GPOCOM        92     A
GPOCOM   *                1           LD   RI,VAL  , VAL = 12/IN[R1],36/CAIH[R2]  GPOCOM  93     A
GPOCOM   *                2           LD   RI,VAL , AND AN   XMTC           GPOCOM        94     A
GPOCOM   *                3           TLD  RI,VAL                           GPOCOM        95     A
GPOCOM   *                4           TLD  RI,VAL  , AND AN XMTC            GPOCOM        96     A
GPOCOM   *                5/6         LDV  RI,VAL , AND XMT/XMTC            CCG0127        8     A
GPOCOM   *                                                                  GPOCOM        97     A
GPOCOM   *         B) THE INTEGER POLYNOMIAL INFORMATION LIST, WHICH CONSISTS OF  GPOCOM  98     A
GPOCOM   *         A POLYNOMIAL INFORMATION WORD ( *PI* ), FOLLOWED BY THE  GPOCOM       99     A
GPOCOM   *         INCREMENT FORMULA AND THE POLYNOMIAL FORMULA.  A FORMULA  GPOCOM     100     A
GPOCOM   *         CONSISTS OF TWO HEADER WORDS ( *IA* AND *IB* ), FOLLOWED  GPOCOM     101     A
GPOCOM   *         BY INSTRUCTIONS ( 4 WORDS/INST ) WHICH DEFINE IT.        GPOCOM       102     A
GPOCOM                                                                      GPOCOM       103     A
GPOCOM             DESCRIBE PI.,60                                          GPOCOM       104     A
GPOCOM    MIP      DEFINE 1          MARKED *IP*                            GPOCOM       105     A
GPOCOM             DEFINE 5                                                 GPOCOM       106     A
GPOCOM    DFI      DEFINE 18         INDEX TO DIFFERENCE FORMULA IN *IIT*   GPOCOM       107         I
         -FCC0514
GPOCOM    CN       DEFINE 18         CLASS NUMBER OF SET OF SIMILAR *IP"S*  FCC0514       20     A
GPOCOM    CA       DEFINE 18         CA OF IP THIS *IP* IS SIMILAR TO       GPOCOM       108     A
GPOCOM    LF       DEFINE 18         LINK FOWARD ( REL TO *O.IIT* )         GPOCOM       109     A
GPOCOM                                                                      GPOCOM       110     A
GPOCOM             DESCRIBE IA.,60                                          GPOCOM       111     A
GPOCOM    NWD      DEFINE 12         P(N.WDS IN FORMULA) , INCLUDES HEADER WDS  GPOCOM   112     A
GPOCOM             DEFINE 12                                                GPOCOM       113     A
GPOCOM    CA       DEFINE 18         CA OF *IP*                             GPOCOM       114     A
GPOCOM    UDI      DEFINE 18         *UDI* INDEX OF INTEGER VARIABLE *LD*   GPOCOM       115     A
GPOCOM                                                                      GPOCOM       116     A
GPOCOM             DESCRIBE IB.,60                                          GPOCOM       117     A
GPOCOM    NLD      DEFINE 12         P(N.LD"S IN FORMULA)                   GPOCOM       118     A
GPOCOM             DEFINE 30                                                GPOCOM       119     A
GPOCOM    ILD      DEFINE 18         *RI* OF *LD* OF *RD* VAR IN EXPRESSION GPOCOM       120     A
GPOCOM                                                                      GPOCOM       121     A
GPOCOM   *         C) A LIST OF POLYNOMIAL INCREMENT VALUES AT THE POINT    GPOCOM       122     A
GPOCOM   *         OF A *RD*.                                               GPOCOM       123     A
GPOCOM   *         THIS LIST CONSISTS OF 2 WORDS / ENTRY, TERMINATED BY A 0 WORD  GPOCOM  124     A
GPOCOM   *                                                                  GPOCOM       125     A
GPOCOM   *         WORD 1 = 60/CA OF *TST* THAT IS BEGIN INCREMENTED        GPOCOM       126     A
GPOCOM   *         WORD 2 = 12/TYPE,48/VAL OF INCREMENT , AS IN LIST A .    GPOCOM       127     A
GPOCOM   *                                                                  GPOCOM       128     A
GPOCOM   *         THE FORMAT OF THE LINK WORD OF THE *RD* IS               GPOCOM       129     A
GPOCOM   *                12/P(N.PI),12/,18/STC,18/INDEX TO LIST IN *IIT*  GPOCOM       130     A
GPOCOM    COMMON   TITLE  GPO/GRA COMMON VARIABLES                          GPOCOM       131     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPOCOM

```
GPOCOM   *         THE FOLLOWING LOCATIONS ARE SHARED BETWEEN *GPO* AND *GRA*          GPOCOM      132    A
GPOCOM                                                                                 GPOCOM      133    A
GPOCOM             USE    /GPOGRA/                                                     GPOCOM      134    A
GPOCOM    O.UDT    BSS    1           FWA OF *UDI* IN LOW CORE                         GPOCOM      135    A
GPOCOM    L.UDT    BSS    1           LENGTH                                           GPOCOM      136    A
GPOCOM    O.MVL    BSS    1           FWA OF MARKED VAR LIST ( IN *UDT* )              GPOCOM      137    A
GPOCOM                                                                                 GPOCOM      138    A
GPOCOM   *         ADDRESS"S OF SPECIAL BIT VECTORS                                    GPOCOM      139    A
GPOCOM                                                                                 GPOCOM      140    A
GPOCOM    DVA      BSS    1           REGION *DEF* VECTOR                              GPOCOM      141    A
GPOCOM    LUV      BSS    1           LOOP USEAGE VECTOR                               GPOCOM      142    A
GPOCOM    UVA      BSS    1           REGION *USE* VECTOR                              GPOCOM      143    A
GPOCOM    LXA      BSS    1           REGION *LIVE EXIT* VECTOR                        GPOCOM      144    A
GPOCOM                                                                                 GPOCOM      145    A
GPOCOM    LEA      BSS    1           REGION *LIVE ENTRY* VECTOR                       GPOCOM      146    A
GPOCOM    MDA      BSS    1           *MOVABLE DEFS*                                   GPOCOM      147    A
GPOCOM    SVA      BSS    1           SCRATCH VECTOR                                   GPOCOM      148    A
GPOCOM    N.SBV    EQU    *-DVA       N. SPECIAL BIT VECTORS                           GPOCOM      149    A
GPOCOM   *                       WSC                                                   GPOCOM      150    A
GPOCOM    M.PL     CON    100B        MAX *PIT* LENGTH                                 GPOCOM      151    A
GPOCOM                                                                                 GPOCOM      152    A
GPOCOM    EMF      CON    2           FP/LVL EXPANSION MODE ( *CBB* ARGUMENT )         GPOCOM      153    A
GPOCOM    ITL      BSS    1           INITIAL *TET* LENGTH                             GPOCOM      154    A
GPOCOM    MRA      VFD    36/,24/-0   MACHINE REGISTERS AVAILABLE ( 8/X,8/A,8/B )      GPOCOM      155    A
GPOCOM    LGL      BSS    1           LP GEN LBL, LL=LP LBL 42/GEN LBL-LL,18/LL        FCC0909       7    A
GPOCOM                                                                                 GPOCOM      156    A
GPOCOM   *                          ?FWA OF CLEARED CELLS                              GPOCOM      157    A
GPOCOM    N.HB     BSS    1           N. HOLDING BLOCKS IN LOOP ( =0 IF INNERMOST      GPOCOM      158    A
GPOCOM    XRF      BSS    1           "0 IF LOOP CONTAINS EXTERNAL REFS                GPOCOM      159    A
GPOCOM                                                                                 GPOCOM      160    A
GPOCOM    LCV      BSS    1           18/UDI,36/CAIH OF LOOP CONTROL VAR               GPOCOM      161    A
GPOCOM             BSS    1           REGNO THAT IT IS IN                              GPOCOM      162    A
GPOCOM    LUL      BSS    1           LOOP UPPER LIMIT IN *IIT A* FORMAT               GPOCOM      163    A
GPOCOM             BSS    1           REGNO THAT IT IS IN                              GPOCOM      164    A
GPOCOM    LIH      BSS    2           IH OF LOOP LABEL / *R2* WORD OF *LAB*            GPOCOM      165    A
GPOCOM    N.LJ     BSS    1           N. JPX"S THAT REF LOOP LABEL                     GPOCOM      166    A
GPOCOM    LTT      BSS    1           LOOP TEST TYPE, =0 IF NO TEST REPLACEMENT        GPOCOM      167    A
GPOCOM   *                            =1 , TEST IS  IF( LCV @ LUL ) GO TO LIH          GPOCOM      168    A
GPOCOM   *                       FXI                                                   GPOCOM      169    A
GPOCOM    O.ENL    BSS    1           FWA OF EXIT NODE LIST IN *BTT*                   GPOCOM      170    A
GPOCOM    N.ENL    BSS    1           N. EXIT NODES                                    GPOCOM      171    A
GPOCOM    EPSI     BSS    1           EPILOGUE POST STORE BIT VECTOR INDEX             GPOCOM      172         I
         -FCC0985
GPOCOM   *                            IN *RXI*, 0 IF NONE.                             GPOCOM      173         I
         -FCC0985
GPOCOM    EPSI     BSS    1           N. EXIT NODES THAT ARE FALL THROUGH              FCC0985       6    A
GPOCOM   *                       CBB                                                   GPOCOM      174    A
GPOCOM    MAXW     BSS    1           MAX *REG WIDTH* OF REGION                        GPOCOM      175    A
GPOCOM    N.EBB    BSS    1           N. EXTENDED BASIC BLOCKS IN REGION               GPOCOM      176    A
GPOCOM                                                                                 GPOCOM      177    A
GPOCOM    PL       BSS    1           *PIT* LENGTH AT START OF BLOCK                   GPOCOM      178    A
GPOCOM    LBN      BSS    1           MAX PROGRAM BLOCK NUMBER CODED ( TET PACK )      GPOCOM      179         I

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
            LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPOCOM

            -FCC0675
GPOCOM     STP      BSS    1           "0 IF SOME *IP* IS A FUNCTION OF THE*LCV*          GPOCOM       180     A
GPOCOM     N.CLC    EQU    *-N.HB     'END OF ZEROED CELLS REGION                         GPOCOM       181     A
GPOCOM              USE    0                                                              GPOCOM       182     A

         SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - GPOCOM

            IDENTIFIER                TOTAL    ACTIVE

            GPOCOM                      182       176
            FCC0514                       1         1
            FCC0909                       1         1
            FCC0985                       2         2
            CCGNC                         1         1
            CCG0127                       1         1

            LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN UDTBLD

UDTBLD     *COMDECK  UDTBLD                                                              UDTBLD         1     A
UDTBLD      UDTBLD   CTEXT  UDTBLD - USE/DEF TABLE FORMAT DEFINITIONS                    UDTBLD         2     A
UDTBLD                                                                                   UDTBLD         3     A
UDTBLD     **        UA. - FORMAT OF USE/DEF INDEX TABLE DURING CHAINING PHASE          UDTBLD         4     A
UDTBLD                                                                                   UDTBLD         5     A
UDTBLD              DESCRIBE UA.,60                                                      UDTBLD         6     A
UDTBLD              DEFINE 5                                                             UDTBLD         7     A
UDTBLD      CR      DEFINE 1                                                             UDTBLD         8     A
UDTBLD      HL      DEFINE 18          HASH LINK                                         UDTBLD         9     A
UDTBLD      CA      DEFINE 18          BIAS ( IF CR " 0, THEN CA = -0 )                  UDTBLD        10     A
UDTBLD      IH      DEFINE 18          BASE ( SYMBOL ORDINAL )                           UDTBLD        11     A
UDTBLD                                                                                   UDTBLD        12     A
UDTBLD      CAIH    DEQU   IH,36                                                         UDTBLD        13     A
UDTBLD      UD      SPACE  2                                                             UDTBLD        14     A
UDTBLD     **       UD. - FORMAT OF USE/DEF TABLE AFTER CHAINING                         UDTBLD        15     A
UDTBLD     *                                                                             UDTBLD        16     A
UDTBLD     *        ENTRY  0 - SPOIL ENTRY ( NO OPTIMIZATION POSSIBLE )                  UDTBLD        17     A
UDTBLD     *               1 - LOAD ONLY VARIABLES ( A0+K )                              UDTBLD        18     A
UDTBLD     *                                                                             UDTBLD        19     A
UDTBLD     *        WORD 2 = 12/P(BITN),30/,18/WI ( WORD INDEX )                         UDTBLD        20     A
UDTBLD                                                                                   UDTBLD        21     A
UDTBLD              DESCRIBE UD.,60                                                      UDTBLD        22     A
UDTBLD      DEFR    DEFINE 1           DEFINED IN REGION                                 UDTBLD        23     A
UDTBLD      DEF     DEFINE 1           IH/CA DEFINED IN BLOCK                            UDTBLD        24     A
UDTBLD      DBU     DEFINE 1           DEFINITION BEFORE USE                             UDTBLD        25     A
UDTBLD      USE     DEFINE 1           USE                                               UDTBLD        26     A
UDTBLD      ^RD     DEFINE 1           NOT RECURSIVELY DEFINED                           UDTBLD        27     A
UDTBLD      IST     DEFINE 1           INTERFERING STORE ( RJX, ETC. )                   UDTBLD        28     A
UDTBLD                                                                                   UDTBLD        29     A
UDTBLD      STC     DEFINE 9           STORE COUNT                                       UDTBLD        30     A
UDTBLD      RF      DEFINE 1           SHORT USAGE IN A LOOP                             FCC1504X       7     A
UDTBLD      FP      DEFINE 1           IH/CA IS A F.P.                                   UDTBLD        31     A
UDTBLD              DEFINE P.FP-P.COM-1                                                  UDTBLD        32          I
            -CCG1

            0         1         2         3         4         5         6         7         8
            12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN UDTBLD

UDTBLD     COM      DEFINE 1           IH/CA IS IN COMMON                          UDTBLD        33     A
UDTBLD              DEFINE 6-P.FP+P.COM                                            UDTBLD        34          I
          -FCC1504X
UDTBLD              DEFINE 5-P.FP+P.COM                                            FCC1504X       8          I
          -CCG1
UDTBLD              DEFINE 4                                                       CCG1           1     A
UDTBLD     BMI      DEFINE 18          INDEX TO BASE MEMBER ( CM " 0 )             UDTBLD        35     A
UDTBLD     CM       DEFINE 1           CLASS MEMBER ( NON BASE MEMBER OF EQV CLASS ) UDTBLD      36     A
UDTBLD     CR       DEFINE 1           CLASS REPRESENTATIVE                        UDTBLD        37     A
UDTBLD     ECL      DEFINE 18          EQUIV CHAIN LINK ( TO NEXT ON CHAIN )       UDTBLD        38     A
UDTBLD                                                                             UDTBLD        39     A
UDTBLD     DEFB     DEQU   DEF,2       DEF BITS ( REGION AND BLOCK )               UDTBLD        40     A
UDTBLD     MV       DEQU   USE,3       MARKED VARIABLE ( USE, DBU, DEF )           UDTBLD        41     A
UDTBLD     CB       DEQU   CR,2        CLASS BITS                                  UDTBLD        42     A
UDTBLD     SMS      DEQU   STC,2       STORE MOTION SPOIL FOR VARS DEF IN INNERLP  UDTBLD        43     A
UDTBLD     R2       SPACE  2                                                       UDTBLD        44     A
UDTBLD    **        R2. - FORMAT OF *R2* WORD OF *BOS* OF BLOCK WRITTEN TO MASS    UDTBLD        45     A
UDTBLD    *                STORAGE                                                 UDTBLD        46     A
UDTBLD                                                                             UDTBLD        47     A
UDTBLD              DESCRIBE R2.,60                                                UDTBLD        48     A
UDTBLD     UXR      DEFINE 1           =1 IF USER EXT REFS IN BLOCK                UDTBLD        49     A
UDTBLD              DEFINE 17                                                      UDTBLD        50          I
          -CCG21
UDTBLD     LONG     DEFINE 1           =1 IF INDEX VARIABLE IS LONG ( 21 BIT ARIT  CCG21          5     A
UDTBLD              DEFINE 16                                                      CCG21          6     A
UDTBLD     MRA      DEFINE 24          MACHINE REGISTERS AVAIL                     UDTBLD        51     A
UDTBLD     TXTL     DEFINE 18          *TXT* LENGTH OF BLOCK                       UDTBLD        52     A
UDTBLD                                                                             UDTBLD        53     A
UDTBLD              ENDX                                                           UDTBLD        54     A

          SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - UDTBLD

              IDENTIFIER                TOTAL    ACTIVE

              UDTBLD                       54        51
              FCC1504X                      2         1
              CCG1                          1         1
              CCG21                         2         2

              LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

MIO       *DECK     MIO                                                            MIO            1     A
MIO                 IDENT  MIO                                                     MIO            2     A
MIO        MIO      TITLE  MIO - MASS STORAGE RANDOM I/O                           MIO            3     A
MIO       *CALL     SSTCALL                                                        MIO            4     A
MIO        B=MIO    RPVDEF                                                         MIO            5     A
MIO                 PASS2TM                                                        MIO            6          I
          -CCG1
MIO       *IF       DEF,MIO,1                                                      MIO            7     A
MIO                 TRACER (WMB,RTB,DMB)                                           MIO            8     A
MIO                                                                                MIO            9     A
MIO       **        MIO - MASS STORAGE I/O ROUTINES                                MIO           10     A

              0         1         2         3         4         5         6         7         8
              12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO      *                                                                  MIO          11     A
MIO      *         THIS SET OF SUBROUTINES HANDLES DYNAMIC STORAGE ALLOCATION  MIO      12     A
MIO      *         FOR EXTENDED BASIC BLOCKS DURING OPT=2 PROCESSING.  BLOCKS  MIO      13     A
MIO      *         ARE PAGED FROM LCM/ECS OR DISK INTO AN SCM TABLE *BLK* AS THEY  MIO  14     A
MIO      *         ARE REQUESTED.  THE RANDOM INDEX WORDS IN *BIT* AND THE BLOCK  MIO  15     A
MIO      *         ADDRESS WORDS IN *BST* ARE UPDATED FOR EACH BLOCK THAT IS  MIO  16     A
MIO      *         MOVED.  THEY CONTAIN CURRENT POSITIONAL INFORMATION INCLUDING  MIO  17     A
MIO      *         THE SCM FWA, DISK ADDRESS, LCM/ECS ADDRESS, LENGTH, AND STATUS  MIO  18     A
MIO      *         BITS FOR EACH BLOCK.                                       MIO      19     A
MIO      *         SPACE IN *BLK* AND LCM/ECS IS ALLOCATED USING KNUTHS BOUNDARY  MIO  20     A
MIO      *         TAG METHOD.  IT REQUIRES HEADER AND TRAILER WORDS FOR EVERY  MIO  21     A
MIO      *         BLOCK.  THE LINKED LIST OF AVAILABLE SPACE IS KEPT IN THE  MIO  22     A
MIO      *         BLOCK TRAILER WORDS.  THE ADVANTAGE OF THIS TECHNIQUE IS THAT  MIO  23     A
MIO      *         ESSENTIALLY A FIXED AMOUNT OF TIME IS REQUIRED TO FREE AN AREA  MIO  24     A
MIO      *         AND COLLAPSE IT WITH ANY ADJACENT FREE AREA, WHILE OTHER  MIO  25     A
MIO      *         METHODS WOULD REQUIRE A LIST SEARCH.  IT ALSO SIMPLIFIES  MIO  26     A
MIO      *         GARBAGE COLLECTION PROCEDURES.  THE METHOD  IS DESCRIBED BY  MIO  27     A
MIO      *         GRIES IN "COMPILER CONSTRUCTION FOR DIGITAL COMPUTERS",  MIO  28     A
MIO      *         SECTION 8.10.                                              MIO      29     A
MIO                                                                          MIO      30          I
         -F1037CC
MIO       #MD      IFLT   CT.CPU,75                                          MIO      31          I
         -S3143FCA -F1037CC
MIO       #MD      IFLT   CT.CPU,76                                          S3143FCA  3          I
         -F1037CC
MIO                                                                          MIO      32          I
         -F1037CC
MIO       RXX,Q    OPDEF  I,J,Q                                              MIO      33          I
         -F1037CC
MIO                R=     A0,Q                                               MIO      34          I
         -F1037CC
MIO                R=     X0,X_J                                             MIO      35          I
         -F1037CC
MIO      +         RE     B1                                                 MIO      36          I
         -F7540FT  -F1037CC
MIO      -         EQ     ERRECSR                                            MIO      37          I
         -F7540FT  -F1037CC
MIO                RL     B1                                                 F7540FT  37          I
         -F1037CC
MIO                SA.I   A0                                                 MIO      38          I
         -F1037CC
MIO       R        ERRPL  I-6         OPERAND NOT IN LD REG                  MIO      39          I
         -F1037CC
MIO                ENDM                                                      MIO      40          I
         -F1037CC
MIO                                                                          MIO      41          I
         -F1037CC
MIO       WXX,Q    OPDEF  I,J,Q                                              MIO      42          I
         -F1037CC
MIO                R=     A0,Q                                               MIO      43          I
         -F1037CC
MIO                R=     X0,X_J                                             MIO      44          I

         0        1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
                -F1037CC
MIO                     SA.I    A0                                                      MIO          45          I
                -F1037CC
MIO             +       WE      B1                                                      MIO          46          I
                -F7540FT -F1037CC
MIO             -       EQ      ERRECSW                                                 MIO          47          I
                -F7540FT -F1037CC
MIO                     WL      B1                                                      F7540FT      38          I
                -F1037CC
MIO              R      ERRMI   I-6         OPERAND NOT IN ST REG                       MIO          48          I
                -F1037CC
MIO                     ENDM                                                            MIO          49          I
                -F1037CC
MIO                                                                                     MIO          50          I
                -F1037CC
MIO              BLK.SIZ EQU     377B                                                    FCC0448      13          I
                -F1037CC
MIO              #MD    ELSE                                                             MIO          51          I
                -F1037CC
MIO                                                                                     MIO          52          I
                -F1037CC
MIO              RXX,Q  OPDEF   I,J,Q                                                   MIO          53          I
                -F1037CC
MIO                     RX.I    X.J                                                     MIO          54          I
                -F1037CC
MIO                     ENDM                                                            MIO          55          I
                -F1037CC
MIO                                                                                     MIO          56          I
                -F1037CC
MIO              WXX,Q  OPDEF   I,J,Q                                                   MIO          57          I
                -F1037CC
MIO                     WX.I    X.J                                                     MIO          58          I
                -F1037CC
MIO                     ENDM                                                            MIO          59          I
                -F1037CC
MIO                                                                                     MIO          60          I
                -F1037CC
MIO              BLK.SIZ EQU     1777B                                                   FCC0448      14          I
                -F1037CC
MIO              #MD    ENDIF                                                            MIO          61          I
                -F1037CC
MIO                     IF      -DEF,.MI,1
>>>>            F1037CC 18                                                              F1037CC       5     A
MIO              .MI    EQU     0
>>>>            F1037CC 19                                                              F1037CC       6     A
MIO
>>>>            F1037CC 20                                                              F1037CC       7     A
MIO              #MD    IFGE    CT.CPU,76
>>>>            F1037CC 21                                                              F1037CC       8     A
MIO              #MD    IFNE    .MI,1
>>>>            F1037CC 22                                                              F1037CC       9     A
MIO

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
>>>>     F1037CC 23                                                                F1037CC       10     A
MIO       RXX,Q    OPDEF  I,J,Q
>>>>     F1037CC 24                                                                F1037CC       11     A
MIO                RX.I   X.J
>>>>     F1037CC 25                                                                F1037CC       12     A
MIO                ENDM
>>>>     F1037CC 26                                                                F1037CC       13     A
MIO
>>>>     F1037CC 27                                                                F1037CC       14     A
MIO       WXX,Q    OPDEF  I,J,Q
>>>>     F1037CC 28                                                                F1037CC       15     A
MIO                WX.I   X.J
>>>>     F1037CC 29                                                                F1037CC       16     A
MIO                ENDM
>>>>     F1037CC 30                                                                F1037CC       17     A
MIO
>>>>     F1037CC 31                                                                F1037CC       18     A
MIO       BLK.SIZ  EQU    1777B
>>>>     F1037CC 32                                                                F1037CC       19     A
MIO       #MD      ELSE
>>>>     F1037CC 33                                                                F1037CC       20     A
MIO       RXX,Q    OPDEF  I,J,Q
>>>>     F1037CC 34                                                                F1037CC       21     A
MIO                R=     A0,Q
>>>>     F1037CC 35                                                                F1037CC       22     A
MIO                R=     X0,X_J
>>>>     F1037CC 36                                                                F1037CC       23     A
MIO                RL     B1
>>>>     F1037CC 37                                                                F1037CC       24     A
MIO                SA.I   A0
>>>>     F1037CC 38                                                                F1037CC       25     A
MIO       R        ERRPL  I-6         OPERAND NOT IN LD REG
>>>>     F1037CC 39                                                                F1037CC       26     A
MIO                ENDM
>>>>     F1037CC 40                                                                F1037CC       27     A
MIO
>>>>     F1037CC 41                                                                F1037CC       28     A
MIO       WXX,Q    OPDEF  I,J,Q
>>>>     F1037CC 42                                                                F1037CC       29     A
MIO                R=     A0,Q
>>>>     F1037CC 43                                                                F1037CC       30     A
MIO                R=     X0,X_J
>>>>     F1037CC 44                                                                F1037CC       31     A
MIO                SA.I   A0
>>>>     F1037CC 45                                                                F1037CC       32     A
MIO                WL     B1
>>>>     F1037CC 46                                                                F1037CC       33     A
MIO       R        ERRMI  I-6         OPERAND NOT IN ST REG
>>>>     F1037CC 47                                                                F1037CC       34     A
MIO                ENDM
>>>>     F1037CC 48                                                                F1037CC       35     A
MIO

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
>>>>     F1037CC 49                                                          F1037CC       36     A
MIO       BLK.SIZ  EQU    377B
>>>>     F1037CC 50                                                          F1037CC       37     A
MIO       #MD      ENDIF
>>>>     F1037CC 51                                                          F1037CC       38     A
MIO                                                                          F7540FT       39     A
MIO       #MD      IFNE   CT.7X*100D+CT.CPU,76D                             F7540FT       40           I
         -F2950CC
MIO       #MD      IF     -MIC,HF.E                                         F2950CC        5     A
MIO                                                                          F7540FT       41     A
MIO                PURGDEF RLB                                              F7540FT       42     A
MIO                PURGDEF WLB                                              F7540FT       43     A
MIO       RLB      OPDEF  I                                                 F7540FT       44     A
MIO      +         RL     B.I+0                                             F7540FT       45     A
MIO      -         EQ     ERRECSR                                           F7540FT       46     A
MIO                ENDM                                                     F7540FT       47     A
MIO                                                                          F7540FT       48     A
MIO       WLB      OPDEF  I                                                 F7540FT       49     A
MIO      +         WL     B.I+0                                             F7540FT       50     A
MIO      -         EQ     ERRECSW                                           F7540FT       51     A
MIO                ENDM                                                     F7540FT       52     A
MIO                                                                          F7540FT       53     A
MIO       RE       OPSYN  ERR                                               F7540FT       54     A
MIO       WE       OPSYN  ERR                                               F7540FT       55     A
MIO                                                                          F7540FT       56     A
MIO       #MD      ELSE
>>>>     F1037CC 53                                                          F1037CC       39     A
MIO                ERRNZ  .MI         SHOULD BE 7600 ONLY (NOT .MI)
>>>>     F1037CC 54                                                          F1037CC       40     A
MIO
>>>>     F1037CC 55                                                          F1037CC       41     A
MIO       #MD      ENDIF                                                    F7540FT       57     A
MIO                                                                          MIO           62     A
MIO                TABLES BLK,TXT,BIT,BST,SEQ                               MIO           63           I
         -CCG1
MIO                                                                          MIO           64           I
         -CCG1
MIO                EXT    BSI,MAX.RL                                        MIO           65           I
         -CCG1
MIO                ECHO   1,Z=(BIT,BLK,BST,SEQ,TXT)                         CCG1           2           I
         -CCGNC
MIO                EXT    O.Z,L.Z                                           CCG1           3           I
         -CCGNC
MIO                TABLES (BIT,BLK,BST,SEQ,TXT)                             CCGNC        481     A
MIO                EXT#   (PUNT,MOREFL,BSI,TO,TOB)                         CCGNC        482     A
MIO                EXT    CP.AFLL                                           CCGNC        483     A
MIO       LCM.FL   EQU    CP.AFLL                                           CCGNC        484     A
MIO                                                                          MIO           66     A
MIO                USE    //                                                MIO           67           I
         -CCG18
MIO                BSS    100B                                              MIO           68           I
         -CCG18

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO       FWAB     BSS    1                                                  MIO          69        I
         -CCG18
MIO                USE    *                                                  MIO          70        I
         -CCG18
MIO                EXT    F.MEM                                              CCG18        81    A
MIO       FWAB     EQU    F.MEM+100B                                         CCG18        82    A
MIO                SPACE  4,8                                                MIO          71    A
MIO      *         GLOBAL FLAGS                                              MIO          72        I
         -CCG11
MIO                                                                          MIO          73        I
         -CCG11
MIO       O.LCM    ENTRY. 0          NEXT AVAILABLE LCM ADDRESS              MIO          74        I
         -CCG7
MIO       MAX.BLK  ENTRY. 0          MAXIMUM SIZE OF *BLK*                   MIO          75        I
         -CCG7
MIO       O.LCM    BSS    1          NEXT AVAIL LCM ADDRESS                  CCG7         80    A
MIO                                                                          MIO          76    A
MIO      *         LOCAL VARIABLES                                           MIO          77    A
MIO                                                                          MIO          78    A
MIO                ENTRY MAX.LCM                                             CCGA012       5    A
MIO       MAX.LCM  DATA   200000B    MAX LCM COMPILER USES                   MIO          79    A
MIO       O.DISK   DATA   1          NEXT AVAILABLE RA ON DISK               MIO          80    A
MIO       ADT      BSSZ   1          ADDRESS TYPE, 0=DISK, 1=LCM             MIO          81    A
MIO       O.RRB    BSSZ   1          FWA OF RANDOM READ BUFFER ( RRB )       MIO          82    A
MIO       O.RWB    VFD    42/,18/FWAB FWA OF RANDOM WRITE BUFFER             MIO          83    A
MIO       RLEN     BSSZ   1          LENGTH OF CURRENT RECORD IN WRITE BUFFER MIO         84    A
MIO       N.RRB    BSSZ   1          RA OF RECORD IN RRB                     MIO          85    A
MIO       MX.AVS   ENTRY. 0          MAXIMUM AVAILABLE SPACE IN *BLK*        MIO          86        I
         -CCGNC
MIO       MX.AVS   ENTRY. 0,#        MAXIMUM AVAILABLE SPACE IN *BLK*        CCGNC       485    A
MIO       MX.AVL   BSSZ   1          MAXIMUM AVAILABLE SPACE IN LCM          MIO          87    A
MIO       LAS      BSSZ   1          HEAD OF BLK LIST OF AVAILABLE SPACE     MIO          88    A
MIO       LAL      BSSZ   1          HEAD OF LCM LIST OF AVAILABLE SPACE     MIO          89    A
MIO                                                                          MIO          90    A
MIO       MIN.AB   EQU    40B        MIN SIZE BLOCK ADDED TO LAS             MIO          91    A
MIO       MIN.AL   EQU    100B       MIN SIZE BLOCK ADDED TO LAL             MIO          92    A
MIO       LCM.XL   EQU    20B        EXTRA LENGTH ADDED TO LCM BLOCKS        MIO          93    A
MIO                                                                          MIO          94    A
MIO                LIST   X                                                  MIO          95    A
MIO      *CALL     IOTBLD                                                    MIO          96    A
MIO       DEBUG    TITLE  DEBUGGING FACILITIES                              MIO          97    A
MIO      **        PRNTM - PRINT OUT SELECTED STRUCTURES                     MIO          98    A
MIO      *                                                                   MIO          99    A
MIO      *         PRNTM  LAB,NAM,ARG,RLS                                    MIO         100    A
MIO                                                                          MIO         101    A
MIO       PRNTM    MACRO  LAB,NAM,ARG,RLS                                    MIO         102    A
MIO                LOCAL  APL                                                MIO         103    A
MIO       O        IF     DEF,/DEBUG/LAB                                     MIO         104    A
MIO      +         RJ     CPR                                                MIO         105    A
MIO      -         VFD    30/APL                                             MIO         106    A
MIO                USE    DEBUG                                              MIO         107    A
MIO       APL      CON    10H LAB                                            MIO         108    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO       N        MICRO  2,1,/ARG/                                          MIO          109     A
MIO                VFD    30/=XSAV.X+"N"                                     MIO          110          I
         -FEAT155F
MIO                VFD    30/=XSV=X+"N"                                      FEAT155F     256     A
MIO                VFD    30/Q.NAM                                           MIO          111     A
MIO                USE    *                                                  MIO          112     A
MIO       O        ENDIF                                                     MIO          113     A
MIO                ENDM                                                      MIO          114     A
MIO                                                                          MIO          115     A
MIO       .T       IFNE   TEST,0                                 * TEST MODE *  MIO       116     A
MIO                                                                          MIO          117     A
MIO                USE    DEBUG                                              FCC0415        5     A
MIO       .1       SET    1                                                  MIO          118     A
MIO                ECHO   2,X=(RIW,BAW,BHW,BTW,BIT,BST,BLK)                  MIO          119     A
MIO       Q.X      EQU    .1                                                 MIO          120     A
MIO       .1       SET    .1+1                                               MIO          121     A
MIO                                                                          MIO          122     A
MIO                                                                          MIO          123     A
MIO       CPR      ROUTINE                                                   MIO          124     A
MIO                RJ     =XSAVE=                                            MIO          125          I
         -CCG7
MIO                RJ     =XSVR=                                             CCG7          81     A
MIO                SA1    CPR                                                MIO          126     A
MIO                LX1    30                                                 MIO          127     A
MIO                SA2    X1-1                                               MIO          128     A
MIO                SA1    O.BIT                                              MIO          129     A
MIO                SA4    O.BST                                              MIO          130     A
MIO                SA5    O.BLK                                              MIO          131     A
MIO                SA3    X2+1                                               MIO          132     A
MIO                SX7    X2+1                                               MIO          133     A
MIO                SX6    X2                                                 MIO          134     A
MIO                LX3    30                                                 MIO          135     A
MIO                SA6    CPRA                                               MIO          136     A
MIO                SA7    CPRA+1                                             MIO          137     A
MIO                BX6    X1                                                 MIO          138     A
MIO                BX7    X4                                                 MIO          139     A
MIO                SA6    CPRA+2                                             MIO          140     A
MIO                SA7    CPRA+4                                             MIO          141     A
MIO                BX6    X5                                                 MIO          142     A
MIO                SX7    X3                                                 MIO          143     A
MIO                SA6    CPRA+6                                             MIO          144     A
MIO                SA7    CPRA+9                                             MIO          145     A
MIO                SX6    X3-8                                               MIO          146     A
MIO                SA6    CPRA+8                                             MIO          147     A
MIO                SA1    CPRA                                               MIO          148     A
MIO                CALL   PRNTMIO                                            MIO          149     A
MIO                RJ     =XRESET=                                           MIO          150          I
         -CCG7
MIO                RJ     =XRSR=                                             CCG7          82     A
MIO                EQ     CPR                                                MIO          151     A
MIO                                                                          MIO          152     A
MIO       CPRA     BSS    3                                                  MIO          153     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO                VFD    42/,18/L.BIT                                                  MIO          154     A
MIO                BSS    1                                                             MIO          155     A
MIO                VFD    42/,18/L.BST                                                  MIO          156     A
MIO                BSS    1                                                             MIO          157     A
MIO                VFD    42/,18/L.BLK                                                  MIO          158     A
MIO                BSSZ   3                                                             MIO          159     A
MIO                USE    0                                                             FCC0415        6     A
MIO                                                                                     MIO          160     A
MIO       .T       ENDIF                                        * TEST MODE *           MIO          161     A
MIO       SETFET   SPACE  3,14                                                           MIO          162     A
MIO      **        SETFET - SETUP FET FOR RANDOM READ OR WRITE ( 6000 CIO )              MIO          163     A
MIO                                                                                     MIO          164     A
MIO       SETFET   MACRO  FET,FWA,LEN,RW,NOADD                                           MIO          165     A
MIO                SA1    RFWA                                                          MIO          166     A
MIO                R=     X6,FWA                                                        MIO          167     A
MIO                R=     B2,LEN                                                        MIO          168     A
MIO                BX6    X1+X6                                                         MIO          169     A
MIO                SA6    =XF.FET+1                                                     MIO          170     A
MIO                SX6    X6                                                            MIO          171     A
MIO                SX7    X6+B2                                                         MIO          172     A
MIO                IFC    EQ,/RW/READ/                                                  MIO          173     A
MIO                SA6    A6+B1       I=F                                               MIO          174     A
MIO                SA6    A6+B1       O=F                                               MIO          175     A
MIO                ELSE                                                                 MIO          176     A
MIO                SA7    A6+B1       I=F+LEN                                           MIO          177     A
MIO                SA6    A7+B1       O=F                                               MIO          178     A
MIO                ENDIF                                                                MIO          179     A
MIO                IFC    EQ,/NOADD//,1                                                 MIO          180     A
MIO                SX7    X7+100B                                                       MIO          181     A
MIO                SA7    A6+B1       LIM=F+L+100B                                      MIO          182     A
MIO                ENDM                                                                 MIO          183     A
MIO                                                                                     MIO          184     A
MIO       RFWA     VFD    13/1,29/3,18/0                                                MIO          185     A
MIO                TTL    MIO - MASS STORAGE RANDOM I/O (EXTERNAL ROUTINES)             MIO          186     A
MIO       IMP      TITLE  IMP - INITIALIZE MASS I/O PROCESSING                          MIO          187     A
MIO      **        IMP - INITIALIZE MASS I/O PROCESSING                                 MIO          188     A
MIO      *                                                                              MIO          189         I
         -FCC0415
MIO      *         ENTRY CONDITIONS FOR IMPC ONLY                                       MIO          190         I
         -FCC0415
MIO      *         ENTRY  (X7) = MAX.BLK, MAXIMUM BLOCK SIZE                            MIO          191         I
         -FCC0415
MIO      *                                                                              MIO          192         I
         -FCC0415
MIO                                                                                     MIO          193     A
MIO       IMPA     ENTRY. **                                                            MIO          194         I
         -CCGNC
MIO       IMPA     ENTRY. **,#                                                          CCGNC        486     A
MIO                                                                                     MIO          195     A
MIO      *         INITIALIZE FOR USING LCM                                             MIO          196     A
MIO                                                                                     MIO          197         I
         -F7540FT

         0        1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO                 SX7    1                                                    MIO          198          I
          -F7540FT
MIO                                                                             MIO          199     A
MIO                 SA1    =XLCM.OA                                             CCG7          83          I
          -CCGNC
MIO                 SA1    =XHO$OFLL                                            CCGNC        487     A
MIO                 BX6    X1          O.LCM = LCM.OA                           CCG7          84          I
          -CCGNC
MIO                 BX6    X1          O.LCM = HO$OFLL                          CCGNC        488     A
MIO                 SA6    O.LCM                                                CCG7          85     A
MIO        #ECS     IFNE   CT.ECS,0                                             MIO          200     A
MIO                 SA1    =XLCM.FL                                             MIO          201     A
MIO                 ZR     X1,IMPA1    IF CP.AFLL = 0  */NO ECS AVAILABLE       MIO          202     A
MIO                 SX7    1                                                    F7540FT       58     A
MIO                 SA7    ADT         ADT = 1                                  MIO          203     A
MIO        #ECS     ENDIF                                                       MIO          204          I
          -F7540FT
MIO                                                                             MIO          205          I
          -F7540FT
MIO        IMPA1    BSS    0                                                    MIO          206     A
MIO                                                                             MIO          207          I
          -CCGA012
MIO        #MD      IFLT   CT.CPU,75                                            MIO          208          I
          -S3143FC  -CCGA012
MIO        #MD      IFLT   CT.CPU,76                                            S3143FC       54          I
          -F7540FT  -CCGA012
MIO        #OS      IFNE   .OS,2                                                F7540FT       59          I
          -CCGA012
MIO                 BX6    X1                                                   MIO          209          I
          -CCGA012
MIO                 SA6    MAX.LCM     MAX.LCM = LCM.FL                         MIO          210          I
          -CCGA012
MIO        #MD      ELSE                                                        MIO          211          I
          -F7540FT  -CCGA012
MIO        #OS      ENDIF                                                       F7540FT       60          I
          -CCGA012
MIO                                                                             F7540FT       61     A
MIO        #ECS     ENDIF                                                       F7540FT       62     A
MIO                                                                             F7540FT       63     A
MIO        #MD      IFEQ   CP#RM,7                                              FCC0415        7     A
MIO                 OPENM  =XFI.OPT,I-O,N,WA                                    MIO          212     A
MIO                 STORE  =XFI.OPT,WA=X7    INITIALIZE WA                      MIO          213     A
MIO        #MD      ENDIF                                                       MIO          214     A
MIO                                                                             MIO          215     A
MIO                 PRINT  IMP,(* --LCM.FL,ADT,MAX.LCM = *3Z8),(X1,ADT,MAX.LCM) MIO          216     A
MIO                 EQ     IMPA                                                 MIO          217     A
MIO                                                                             MIO          218          I
          -FCC0415
MIO        IMPB     SPACE  3                                                    FCC0415        8     A
MIO        IMPB     ENTRY. **                                                   MIO          219          I
          -CCGNC
MIO        IMPB     ENTRY. **,#                                                 CCGNC        489     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO                SA4    O.RWB                                                  MIO          220     A
MIO                SA5    RLEN                                                   MIO          221     A
MIO                ZR     X5,IMPB1                                               MIO          222     A
MIO                                                                              MIO          223     A
MIO                CALL   SBD         DUMP LAST PHASE 1 BLOCKS TO DISK           MIO          224     A
MIO                MX7    0                                                      MIO          225     A
MIO                SA7    RLEN        RLEN = 0                                   MIO          226     A
MIO                                                                              MIO          227     A
MIO       IMPB1    SA1    =XF.RLST+1                                             MIO          228         I
         -CCG7
MIO       IMPB1    SA1    =XF.LGO+1                                              CCG7          86         I
         -CCG18
MIO       IMPB1    SA1    =XHO$OBO                                               CCG18         83     A
MIO                SA3    MAX.RL                                                 MIO          229         I
         -CCG1
MIO                SX6    X1                                                     MIO          230         I
         -CCG1
MIO                IX7    X3+X6                                                  MIO          231         I
         -CCG1
MIO                SX7    X1+=XCO.OBL                                            CCG1           4         I
         -CCGNC
MIO                SX7    X1+=XHO$OBL                                            CCGNC        490     A
MIO                SX6    X1                                                     CCG1           5     A
MIO                SA6    O.RRB       O.RRB = FWA OF RLST BUFFER                 MIO          232         I
         -CCG7
MIO                SA6    O.RRB       O.RRB = FWA OF *LGO* BUFFER                CCG7          87         I
         -CCG18
MIO                SA6    O.RRB       O.RRB = FWA OF OPT=2 PAGING BUFFER         CCG18         84     A
MIO                SA7    O.RWB       O.RWB = FWA OF RLST BUFFER + MAX.RL        MIO          233         I
         -CCG1
MIO                SX6    -1                                                     MIO          234         I
         -CCG1
MIO                MX6    59                                                     CCG1           6     A
MIO                SA7    O.RWB       O.RWB = FWA OF RLIST BUFFER + CO.OBL       CCG1           7         I
         -CCG7
MIO                SA7    O.RWB       O.RWB = O.RRB + CO.OBL                     CCG7          88         I
         -CCGNC
MIO                SA7    O.RWB       O.RWB = O.RRB + HO$OBL                     CCGNC        491     A
MIO                SA6    N.RRB       N.RRB = -1                                 MIO          235     A
MIO                PRINT  IMP,(* --O.RRB,O.RWB =*2Z8),(O.RRB,O.RWB)              MIO          236     A
MIO                EQ     IMPB                                                   MIO          237     A
MIO       IMP      SPACE  3,14                                                   MIO          238     A
MIO      **        IMPC - OPT=2 ENTRY TO INITIALIZE *BLK*                        MIO          239     A
MIO      *                                                                       MIO          240     A
MIO      *         ENTRY  (X7) = MAX BLK SPACE                                   MIO          241     A
MIO                                                                              MIO          242     A
MIO       IMPC     ENTRY. **                                                     MIO          243         I
         -CCGNC
MIO       IMPC     ENTRY. **,#                                                   CCGNC        492     A
MIO                SA7    MAX.BLK                                                MIO          244         I
         -CCG7
MIO                RJ     IBS         INITIALIZE *BLK* SPACE                     MIO          245     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO                SA1    O.BLK                                                    MIO          246     A
MIO                SA2    L.BLK                                                    MIO          247     A
MIO                IX6    X1+X2       O.TXT = O.BLK + L.BLK                        MIO          248     A
MIO                SA6    O.TXT                                                    MIO          249     A
MIO                PRINT  IMP,(*  MAX.BLK,O.BLK,L.BLK =*3Z7),(MAX.BLK,B2,B3)       MIO          250     A
MIO                SX7    =XTOB                                                    MIO          251     A
MIO                SA7    =XTO        SET OPT=2 TABLE OVERFLOW EXIT                MIO          252     A
MIO                EQ     IMPC                                                     MIO          253     A
MIO       IMPD     SPACE  2                                                        MIO          254     A
MIO      **        IMPD - OPT=1 ENTRY TO INITIALIZE *BLK*                          MIO          255     A
MIO      *         ALLOCATE BLOCK SPACE, MOVE LOOP BODY TO *BLK*                   MIO          256     A
MIO                                                                                MIO          257     A
MIO       IMPD     ENTRY. **                                                       MIO          258           I
         -CCGNC
MIO       IMPD     ENTRY. **,#                                                     CCGNC        493     A
MIO                SA1    L.TXT                                                    MIO          259     A
MIO                SX6    0                                                        MIO          260     A
MIO                SA6    LAS         LAS = 0                                      MIO          261     A
MIO                SA6    MX.AVS      MX.AVS = 0                                   MIO          262     A
MIO                SX7    X1+200B                                                  MIO          263     A
MIO                RJ     IBS         INITIALIZE *BLK* SPACE                       MIO          264     A
MIO                SA5    L.TXT                                                    MIO          265     A
MIO                ALLOC  BLK,X5+2    ALLOC( BLK , L.TXT+2 )                       MIO          266     A
MIO                SX3    X2+B6       TO = O.BLK + OLD(L.BLK)                      MIO          267     A
MIO                SB3    X2+B6       FTB = TO                                     MIO          268     A
MIO                SA2    O.TXT                                                    MIO          269     A
MIO                SB4    B3+X5       LTB = TO + L.TXT                             MIO          270     A
MIO                SX6    X5                                                       MIO          271     A
MIO                SA6    X2+B1       R2 = TXT + 1;  [R2] = L.TXT                  MIO          272     A
MIO                MOVE   X1,X2,X3    MOVE( L.TXT , O.TXT , TO )                   MIO          273     A
MIO                SX6    X5+2        LT = L.TXT + 2                               MIO          274     A
MIO                SA6    B3-B1       [FTB-1] = BHW(0,0,0,LT)                      MIO          275           I
         -FCC0415
MIO                SX7    4                                                        FCC0415        9     A
MIO                SA7    L.TXT       L.TXT = 4                                    FCC0415       10     A
MIO                LX7    BH.BIP                                                   FCC0415       11     A
MIO                SA6    B4          [LTB] = BTW(0,0,0,LT)                        MIO          276     A
MIO                BX6    X6+X7                                                    FCC0415       12     A
MIO                MX7    0                                                        MIO          277     A
MIO                SA7    A6+B1       [LTB+1] = 0   */ BLK BOUNDARY MARKER         MIO          278     A
MIO                SX7    4                                                        MIO          279           I
         -FCC0415
MIO                SA7    L.TXT       L.TXT = 4                                    MIO          280           I
         -FCC0415
MIO                SA6    B3-B1       [FTB-1]=BHW(0,4,0,LT)                        FCC0415       13     A
MIO                SA1    O.BIT                                                    MIO          281     A
MIO                MX0    1                                                        MIO          282     A
MIO                LX0    1+RI.ICP                                                 MIO          283     A
MIO                LX5    RI.LENP                                                  MIO          284     A
MIO                SX4    B3                                                       MIO          285     A
MIO                BX6    X0+X5                                                    MIO          286     A
MIO                LX4    RI.FWAP                                                  MIO          287     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO                BX7    X4+X6                                                MIO          288     A
MIO                SA7    X1+5        BIT(5) = RIW(IC,LT-2,FTB,0)              MIO          289     A
MIO                EQ     IMPD                                                 MIO          290     A
MIO       IBS      SPACE  2,10                                                 MIO          291     A
MIO      **        IBS - INTIALIZE *FREE* SPACE LIST IN *BLK*                  MIO          292     A
MIO      *                                                                     MIO          293     A
MIO      *         ENTRY  (X7) = LENGTH OF FREE SPACE LIST                     MIO          294     A
MIO                                                                            MIO          295     A
MIO       IBS      ROUTINE                                                     MIO          296     A
MIO                ALLOC  BLK,X7+2    ALLOC( BLK , LEN+2 )                     MIO          297     A
MIO                SA4    RBSA+1      DB = RBSA(2)  */ *RI* WORD FOR DUMMY BLOCK MIO          298     A
MIO                SA5    O.BIT                                                MIO          299     A
MIO                MX6    0                                                    MIO          300     A
MIO                SA6    X2          [BLK] = 0                                MIO          301     A
MIO                BX7    X4                                                   MIO          302     A
MIO                SA7    X5+B1       BIT(1) = DB                              MIO          303     A
MIO                SA7    X5+3        BIT(3) = DB                              MIO          304     A
MIO                SB3    X3-2        LEN = L.BLK - 2                          MIO          305     A
MIO                SB2    X2+B1       FWA = O.BLK + 1                          MIO          306     A
MIO                SA6    B2+B3       BLK(L.BLK-1) = 0  */ TERMINAL BOUNDARY MARK MIO          307     A
MIO                CALL   ABB         CALL  ABB( FWA , LEN ) */ INIT FREE SPACE MIO          308     A
MIO                EQ     IBS                                                  MIO          309     A
MIO       WMB      TITLE  WMB - WRITE MODIFIED BLOCK                           MIO          310     A
MIO      **        WMB/WFB - WRITE MODIFIED BLOCK TO *BLK*                     MIO          311     A
MIO      *                                                                     MIO          312     A
MIO      *         ENTRY  (BSI) = *BST* INDEX OF BLOCK                         MIO          313     A
MIO      *                FOR THE *WFB* ENTRY ONLY -                           MIO          314     A
MIO      *                (X1) = NL, NEW LENGTH                                MIO          315     A
MIO      *                (X6) = FWAB , FWA OF BLOCK                           MIO          316     A
MIO      *                                                                     MIO          317     A
MIO      *         EXIT   BLOCK COPIED TO *BLK*                                MIO          318     A
MIO      *                *BST*, *BIT* ENTRIES UPDATED                         MIO          319     A
MIO      *                                                                     MIO          320     A
MIO      *         CALLS  ASB, MVE=.                                           MIO          321     A
MIO                                                                            MIO          322     A
MIO       WFB      ENTRY. **                                                   MIO          323         I
         -CCGNC
MIO       WFB      ENTRY. **,#                                                 CCGNC        494     A
MIO                SA2    =XOPT2                                              FCC0534        4         I
         -CCG7
MIO                SA2    =XCO.OPT                                            CCG7          89         I
         -CCGNC
MIO                SA2    =XHO$OPT                                            CCGNC        495     A
MIO                SA3    O.BST                                               MIO          324     A
MIO                SA4    BSI                                                 MIO          325         I
         -CCG1
MIO                SA4    =XBSI                                               CCG1           8     A
MIO                SX0    B1                                                  MIO          326         I
         -FCC0534
MIO                SB3    X3                                                  MIO          327     A
MIO                LX0    BA.FWP                                              MIO          328         I
         -FCC0534

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO              SA5    B3+X4       BAW = BST(BSI)                          MIO          329     A
MIO              BX7    X1                                                  MIO          330         I
         -FCC0534
MIO              ZR     X2,WFB1     IF OPT2=0      */FW NOT SET FOR OPT=1   FCC0534        5         I
         -CCG7
MIO              PL     X2,WFB1     IF OPT " 2    */ FW NOT SET FOR OPT=1   CCG7          90     A
MIO              SX0    B1                                                  FCC0534        6     A
MIO              LX0    BA.FWP                                              FCC0534        7     A
MIO              BX5    X0+X5       FW[BAW] = 1   */ INDICATE FINAL WRITE   MIO          331     A
MIO       WFB1   BX7    X1                                                  FCC0534        8     A
MIO              SA6    WFBA                                                MIO          332     A
MIO              SA7    A6+B1                                               MIO          333     A
MIO              LX5    -BA.FWAP                                            MIO          334     A
MIO              SB2    X5-1        FWA = FWA[BAW] - 1                      MIO          335     A
MIO              LX5    BA.FWAP-BA.LENP                                     MIO          336     A
MIO              SB5    X5+2        OL = LEN[BAW] + 2                       MIO          337     A
MIO              LX5    BA.LENP                                             MIO          338     A
MIO              CALL   ASB         FBLK = ASB(FWA,OL,NL)  */ ALLOCATE SPACE MIO         339         I
         -FCC0634
MIO              MI     X5,WFB2     IF IC[BAW]     */ IF IN BLK             FCC0634        5     A
MIO              SB2    B0          FWA = 0                                 FCC0634        6     A
MIO              SB5    B1+B1       OL = 2                                  FCC0634        7     A
MIO       WFB2   CALL   ASB         FBLK = ASB(FWA,OL,NL)   */ ALLOCATE SPACE FCC0634      8         I
         -F2950CC
MIO              SA2    WFB                                                 MIO          340         I
         -F2950CC
MIO       WFB2   SA2    WFB                                                 F2950CC        6     A
MIO              BX7    X2                                                  F2950CC        7     A
MIO              SA7    WMB         [WMB] = [WFB] */ PLUG ENTRY POINT       F2950CC        8     A
MIO              CALL   ASB                                                 F2950CC        9     A
MIO              SA4    WFBA        FSCM = FWAB                             MIO          341     A
MIO              SA1    A4+B1                                               MIO          342     A
MIO              BX7    X2                                                  MIO          343         I
         -F2950CC
MIO              SA7    WMB         [WMB] = [WFB] */ PLUG ENTRY POINT       MIO          344         I
         -F2950CC
MIO              EQ     WMB1                                                MIO          345     A
MIO                                                                         MIO          346     A
MIO       WMB    ENTRY. **                                                  MIO          347         I
         -CCGNC
MIO       WMB    ENTRY. **,#                                                CCGNC        496     A
MIO              SA3    O.BST                                               MIO          348     A
MIO              SA4    BSI                                                 MIO          349     A
MIO              SB3    X3                                                  MIO          350     A
MIO              SA5    B3+X4       BAW = BST(BSI)                          MIO          351     A
MIO              LX5    -BA.FWAP                                            MIO          352     A
MIO              SB2    X5-1        FWA = FWA[BAW] - 1                      MIO          353     A
MIO              LX5    BA.FWAP-BA.LENP                                     MIO          354     A
MIO              SB5    X5+2        OL = LEN[BAW] + 2                       MIO          355     A
MIO              LX5    BA.LENP                                             MIO          356     A
MIO              SA1    L.TXT       NL = L.TXT                              MIO          357     A
MIO              CALL   ASB         FBLK = ASB(FWA,OL,NL)  */ ALLOCATE SPACE MIO         358         I

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

          -FCC0634
MIO                MI     X5,WMB0     IF IC[BAW]     */ IF IN BLK                    FCC0634        9     A
MIO                SB2    B0          FWA = 0                                        FCC0634       10     A
MIO                SB5    B1+B1       OL = 2                                         FCC0634       11     A
MIO       WMB0     CALL   ASB         FBLK = ASB(FWA,OL,NL)   */ ALLOCATE SPACE      FCC0634       12     A
MIO                                                                                 MIO          359          I
          -FCC0415
MIO                SA1    O.BST                                                      FCC0415       14     A
MIO                SA2    BSI                                                        FCC0415       15     A
MIO                IX6    X1+X2                                                      FCC0415       16     A
MIO                SA5    X6          BAW = BST(BSI)                                 FCC0415       17     A
MIO                SA4    O.TXT       FSCM = [O.TXT]                                 MIO          360     A
MIO                SA1    L.TXT       NL = L.TXT                                     MIO          361     A
MIO                                                                                 MIO          362     A
MIO      *         UPDATE *BST*, *BIT* ENTRIES                                      MIO          363     A
MIO                                                                                 MIO          364     A
MIO       WMB1     SA2    O.BIT                                                      MIO          365     A
MIO                PRINT  WMB,(* --FBLK FROM ASB =*Z7),X3                            MIO          366     A
MIO                SB7    X4                        (B7) = FSCM                       MIO          367     A
MIO                MX0    -BA.SAVL                                                   MIO          368     A
MIO                SB2    X2+B1                                                      MIO          369     A
MIO                LX0    BA.SAVP                                                    MIO          370     A
MIO                MX7    1                                                          MIO          371     A
MIO                BX5    -X0*X5                                                     MIO          372     A
MIO                LX7    1+BA.BLKP   BLK[BAW] = 1                                   MIO          373     A
MIO                BX7    X5+X7                                                      MIO          374     A
MIO                LX5    -BA.BIP                                                    MIO          375     A
MIO                SB4    X5          AR = BI[BAW]                                   MIO          376     A
MIO                LX1    BA.LENP     LEN[BAW] = NL                                  MIO          377     A
MIO                SA2    WMBA                                                       MIO          378     A
MIO                SA4    B4+B2       RIW = BIT(AR+1)                                MIO          379     A
MIO                LX3    BA.FWAP     FWA[BAW] = FBLK                                MIO          380     A
MIO                IX6    X1+X3                                                      MIO          381     A
MIO                LX5    BA.BIP                                                     MIO          382     A
MIO                BX7    X6+X7                                                      MIO          383     A
MIO                LX1    RI.LENP-BA.LENP   LEN[RI] = NL                             MIO          384     A
MIO                SA7    A5          BST(BSI) = BAW                                 MIO          385     A
MIO                PRNTM  WMB,BAW,X7                                                 MIO          386     A
MIO                MX7    1                                                          MIO          387     A
MIO                LX3    RI.FWAP-BA.FWAP   FWA[RI] = FBLK                           MIO          388     A
MIO                LX7    1+RI.ICP    IC[RIW] = 1                                    MIO          389     A
MIO                BX4    X2*X4                                                      MIO          390     A
MIO                BX7    X4+X7                                                      MIO          391     A
MIO                IX6    X1+X3                                                      MIO          392     A
MIO                BX7    X6+X7                                                      MIO          393     A
MIO                LX1    -RI.LENP                                                   MIO          394     A
MIO                LX3    -RI.FWAP                                                   MIO          395     A
MIO                SA7    A4          BIT(AR+1) = RIW                                MIO          396     A
MIO                PRNTM  WMB,RIW,X7                                                 MIO          397     A
MIO                                                                                 MIO          398     A
MIO      *         STORE BLOCK HEADER AND TRAILER WORDS, MOVE BLOCK TO *BLK*.       MIO          399     A
MIO                                                                                 MIO          400     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO              SA4    BSI                                                MIO        401    A
MIO              BX7    X5                                                 MIO        402    A
MIO              LX7    59-BH.FWP                                          MIO        403    A
MIO              PL     X7,WMB2     IF ^ FW[BHW]                           MIO        404    A
MIO                                                                        MIO        405    A
MIO              MX6    -BH.PRIL                                           MIO        406    A
MIO              SX4    B1                                                 MIO        407    A
MIO              LX6    BH.PRIP                                            MIO        408    A
MIO              BX5    X5*X6       PRI[BHW] = 0                           MIO        409    A
MIO              LX4    17          BSI = 400000B                          MIO        410    A
MIO       WMB2   SX6    X1+2        L= NL + 2                              MIO        411    A
MIO              LX6    BT.LENP                                            MIO        412    A
MIO              SB6    X3                                                 MIO        413    A
MIO              SA6    B6+X1       [FBLK+NL] = BTW(0,0,0,L)               MIO        414    A
MIO              PRNTM  WMB,BTW,X6                                         MIO        415    A
MIO              LX6    BH.LENP-BT.LENP                                    MIO        416    A
MIO              LX4    BH.BSTP                                            MIO        417    A
MIO              BX6    X6+X4                                              MIO        418    A
MIO              BX7    X5+X6                                              MIO        419    A
MIO              SA7    B6-B1       [FBLK-1] = BHW(0,AR,BSI,L)             MIO        420    A
MIO              PRNTM  WMB,BHW,X7                                         MIO        421    A
MIO              MOVE   X1,B7,X3    MOVE( NL , FSCM , FBLK )               MIO        422    A
MIO              EQ     WMB                                                MIO        423    A
MIO                                                                        MIO        424    A
MIO       WMBA   BFMW   RI,(IC,AT,LCM)                                     MIO        425    A
MIO       WFBA   BSS    2           FWA , NL                               MIO        426    A
MIO       RTB    TITLE  RTB - READ TEXT BLOCK                              MIO        427    A
MIO      **      RTB - READ TEXT BLOCK INTO *BLK*                          MIO        428    A
MIO      *                                                                 MIO        429    A
MIO      *       ENTRY  (BSI) = *BST* INDEX OF BLOCK TO BE READ IN         MIO        430    A
MIO      *                                                                 MIO        431    A
MIO      *       EXIT   (O.SEQ) = FWA OF BLOCK IN *BLK*                    MIO        432    A
MIO      *              (L.SEQ) = LEN OF BLOCK                             MIO        433    A
MIO      *              *BST*, *BIT* ENTRIES UPDATED                       MIO        434    A
MIO      *                                                                 MIO        435    A
MIO      *       CALLS  ASB, GMB.                                          MIO        436    A
MIO                                                                        MIO        437    A
MIO      *       BLOCK IS ALREADY IN *BLK*, SET O.SEQ, L.SEQ               MIO        438    A
MIO                                                                        MIO        439    A
MIO       RTB0   LX3    1+BA.BLKP-BA.FWAP                                  MIO        440    A
MIO              SX6    X3                                                 MIO        441    A
MIO              LX3    BA.FWAP-BA.LENP                                    MIO        442    A
MIO              SA6    O.SEQ       O.SEQ = FWA[BAW]                       MIO        443    A
MIO              SX7    X3+                                                MIO        444    A
MIO              SA7    L.SEQ       L.SEQ = LEN[BAW]                       MIO        445    A
MIO                                                                        MIO        446    A
MIO              PRINT  RTB,(*  O.SEQ, L.SEQ = *2Z10),(O.SEQ,L.SEQ)        MIO        447    A
MIO                                                                        MIO        448    A
MIO       RTB    ENTRY. **                                                 MIO        449         I
         -CCGNC
MIO       RTB    ENTRY. **,#                                               CCGNC      497    A
MIO              SA1    O.BST                                              MIO        450    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO                SA2    BSI                                                    MIO          451     A
MIO                SB2    X1                                                     MIO          452     A
MIO                SA3    X2+B2        BAW = BST(BSI)                            MIO          453     A
MIO                LX3    59-BA.BLKP                                             MIO          454     A
MIO                MI     X3,RTB0      IF BLK[BAW]                               MIO          455     A
MIO                                                                             MIO          456     A
MIO      *         ALLOCATE SPACE IN *BLK*                                      MIO          457     A
MIO                                                                             MIO          458     A
MIO                SA4    O.BIT                                                  MIO          459     A
MIO                LX3    1+BA.BLKP-BA.BIP                                       MIO          460     A
MIO                SB4    X3+B1        I = BI[BAW] + 1                           MIO          461     A
MIO                SB2    B0           FWA = 0                                   MIO          462     A
MIO                SB5    B1+B1        OL = 2                                    MIO          463     A
MIO                SA5    X4+B4        RIW = BIT(I)  */RANDOM INDEX WORD         MIO          464     A
MIO                LX5    -RI.LENP                                               MIO          465     A
MIO                SX1    X5           L = LEN[RIW]                              MIO          466     A
MIO                BX4    X5                                                     MIO          467     A
MIO                LX4    59-RI.ICP+RI.LENP                                      MIO          468     A
MIO                PL     X4,RTB1      IF ^ IC[RIW]                              MIO          469     A
MIO                                                                             MIO          470     A
MIO                LX4    1+RI.ICP-RI.FWAP                                       MIO          471     A
MIO                SB5    B0                                                     MIO          472     A
MIO                SX3    X4           FBLK = FWA[RIW]                           MIO          473     A
MIO                SX6    RBSB                                                   MIO          474     A
MIO                IX7    X3-X6                                                  MIO          475     A
MIO                ZR     X7,RTB2      */IF DUMMY BLOCK                          MIO          476     A
MIO                                                                             MIO          477     A
MIO                SA4    X4-1         BHW = [FBLK-1]                            MIO          478     A
MIO                MX7    -BH.BSTL                                               MIO          479     A
MIO                LX4    -BH.BSTP                                               MIO          480     A
MIO                BX4    X4*X7                                                  MIO          481     A
MIO                BX6    X2+X4        BST[BHW] = BSI                            MIO          482     A
MIO                LX6    BH.BSTP                                                MIO          483     A
MIO                SA6    A4           [FBLK] = BHW                              MIO          484     A
MIO                EQ     RTB2                                                   MIO          485     A
MIO                                                                             MIO          486     A
MIO       RTB1     LX5    RI.LENP                                                MIO          487     A
MIO                CALL   ASB          FBLK = ASB(FWA,OL,L)  */ALLOCATE SPACE    MIO          488     A
MIO                LX5    -RI.LENP                                               MIO          489         I
          -FCC0391
MIO                SB5    1                                                      MIO          490         I
          -FCC0391
MIO                SA5    A5           RIW = BIT(I)  */ RELOAD RANDOM ADDR       FCC0391        7     A
MIO                SB5    1                                                      FCC0391        8     A
MIO                LX5    -RI.LENP                                               FCC0391        9     A
MIO                                                                             MIO          491     A
MIO                PRINT  RTB,(* --FBLK FROM ASB = *Z7),X3                        MIO          492     A
MIO                                                                             MIO          493     A
MIO      *         UPDATE *BST*, *BIT* ENTRIES AND READ BLOCK                   MIO          494     A
MIO                                                                             MIO          495     A
MIO       RTB2     SA4    O.BST                                                  MIO          496     A
MIO                SA2    BSI                                                    MIO          497     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO          SX1    X5                                                  MIO        498    A
MIO          SB2    X4                                                  MIO        499    A
MIO          LX1    BA.LENP     LEN[BAW] = L                            MIO        500    A
MIO          MX0    1                                                   MIO        501    A
MIO          SA2    X2+B2       BAW = BST(BSI)                          MIO        502    A
MIO          LX0    1+BA.BLKP   BLK[BAW] = 1                            MIO        503    A
MIO          SX4    B1                                                  MIO        504    A
MIO          BX1    X0+X1                                               MIO        505    A
MIO          LX4    BA.PRIP     PRI[BAW] = 1                            MIO        506    A
MIO          SX7    X3                                                  MIO        507    A
MIO          LX3    BA.FWAP     FWA[BAW] = FBLK                         MIO        508    A
MIO          BX1    X1+X2                                               MIO        509    A
MIO          IX4    X3+X4                                               MIO        510    A
MIO          SA7    O.SEQ       O.SEQ = FBLK                            MIO        511    A
MIO          PRINT  RTB,(*--O.SEQ =*Z7),X7                              MIO        512    A
MIO          BX6    X1+X4                                               MIO        513    A
MIO          SX7    X5                                                  MIO        514    A
MIO          LX3    RI.FWAP-BA.FWAP                                     MIO        515    A
MIO          SA6    A2          BST(BSI) = BAW                          MIO        516    A
MIO          SA7    L.SEQ       L.SEQ = L                               MIO        517    A
MIO          PRNTM  RTB,BAW,X6                                          MIO        518    A
MIO          PRINT  RTB,(* --L.SEQ =*Z7),L.SEQ                          MIO        519    A
MIO          ZR     B5,RTB      IF IC[RIW]                              MIO        520    A
MIO                                                                    MIO        521    A
MIO          LX0    RI.ICP-BA.BLKP    IC[RIW] = 1                      MIO        522    A
MIO          SA4    RTBA        M1 = MASK [RI](AT,LCM,LEN)              MIO        523    A
MIO          BX0    X0+X3                                               MIO        524    A
MIO          LX5    RI.LENP                                             MIO        525    A
MIO          BX7    X4*X5       TRIW = M1 & RIW                         MIO        526    A
MIO          BX7    X0+X7       FWA[RIW] = FBLK                         MIO        527    A
MIO          LX3    -RI.FWAP                                            MIO        528    A
MIO          SA7    A5          BIT(I) = TRIW                           MIO        529    A
MIO          PRNTM  RTB,RIW,X7                                          MIO        530    A
MIO          CALL   GMB         CALL GMB(FBLK,RIW)  */READ BLOCK        MIO        531    A
MIO          EQ     RTB                                                 MIO        532    A
MIO                                                                    MIO        533    A
MIO     RTBA     BFMW   RI,(AT,LCM,LEN)                                MIO        534    A
MIO     RBS      TITLE  RBS - RELEASE BLOCK SPACE                      MIO        535    A
MIO    **        RBS - RELEASE BLOCK SPACE                             MIO        536    A
MIO    *                                                              MIO        537    A
MIO    *         ENTRY  (BSI) = *BST* INDEX OF BLOCK                   MIO        538    A
MIO    *                                                              MIO        539    A
MIO    *         EXIT   BLOCK REDUCED TO BOS, EOQ                      MIO        540    A
MIO    *                                                              MIO        541    A
MIO    *         CALLS  ABB, ABL.                                      MIO        542    A
MIO                                                                    MIO        543    A
MIO     RBS      ENTRY. **                                             MIO        544         I
       -CCGNC
MIO     RBS      ENTRY. **,#                                           CCGNC      498    A
MIO              PRINT  RBS,(* --RELEASE BLOCK, BSI =*Z7),BSI          MIO        545    A
MIO                                                                    MIO        546    A
MIO    *         UPDATE *BST*, *BIT* ENTRIES                           MIO        547    A

       0         1         2         3         4         5         6         7         8
       12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO                                                                                  MIO        548    A
MIO                 SA1    RBSA        DBAW = [RBSA] */BST ENTRY FOR DUMMY BLOCK     MIO        549    A
MIO                 SA2    A1+1        DRIW = [RBSA+1] */BIT ENTRY FOR DUMMY BLOCK   MIO        550    A
MIO                 SA3    O.BST                                                     MIO        551    A
MIO                 SA4    BSI                                                       MIO        552    A
MIO                 SA5    O.BIT                                                     MIO        553    A
MIO                 SB4    RBSB                                                      MIO        554    A
MIO                 SB2    X3                                                        MIO        555    A
MIO                 SB3    X5+B1                                                     MIO        556    A
MIO                 SA3    B2+X4       BAW = BST(BSI)                                MIO        557    A
MIO                 LX3    -BA.BIP     B = BI[BAW]                                   MIO        558    A
MIO                 SA4    B3+X3       RIW = BIT(B)                                  MIO        559    A
MIO                 SX6    X3                                                        MIO        560    A
MIO                 BX7    X2                                                        MIO        561    A
MIO                 LX6    BA.BIP      BI[DBAW] = B                                  MIO        562    A
MIO                 BX6    X1+X6                                                     MIO        563    A
MIO                 SA6    A3          BST(BSI) = DBAW                               MIO        564    A
MIO                 PRNTM  RBS,BAW,X6                                                MIO        565    A
MIO                 SA7    B3+X3       BIT(B) = DRIW                                 MIO        566    A
MIO                 PRNTM  RBS,RIW,X7                                                MIO        567    A
MIO                 LX3    BA.BIP+59-BA.BLKP                                         MIO        568    A
MIO                 PL     X3,RBS1     IF BLK[BAW] = 0                               MIO        569    A
MIO                                                                                  MIO        570    A
MIO       *         RELEASE STORAGE IN BLK                                          MIO        571    A
MIO                                                                                  MIO        572    A
MIO                 LX3    1+BA.BLKP-BA.FWAP                                         MIO        573    A
MIO                 SB2    X3          FBLK = FWA[BAW]                               MIO        574    A
MIO                 EQ     B2,B4,RBS   IF FBLK = RBSB  */NULL BLOCK                 MIO        575    A
MIO                                                                                  MIO        576    A
MIO                 SB4    B1+B1                                                     MIO        577    A
MIO                 LX3    BA.FWAP-BA.LENP                                           MIO        578    A
MIO                 SB3    X3+B4       L = LEN[BAW] + 2                              MIO        579    A
MIO                 SB2    B2-B1       FBLK = FBLK-1                                 MIO        580    A
MIO                 CALL   ABB         CALL ABB(FBLK,L)                              MIO        581    A
MIO                                                                                  MIO        582    A
MIO       *         RELEASE STORAGE IN LCM                                          MIO        583    A
MIO                                                                                  MIO        584    A
MIO        RBS1     LX4    59-RI.ATP                                                 MIO        585    A
MIO                 PL     X4,RBS      IF AT[RIW] = 0                                MIO        586    A
MIO                                                                                  MIO        587    A
MIO        #ECS     IFNE   CT.ECS,0                                                  MIO        588    A
MIO                 LX4    1+RI.ATP-RI.LCMP                                          MIO        589    A
MIO                 SB2    X4                                                        MIO        590    A
MIO                 SX0    X4                                                        MIO        591    A
MIO                 RX1    X0,RBSC                                                   MIO        592    A
MIO                 LX1    -BH.LENP                                                  MIO        593    A
MIO                 SB3    X1                                                        MIO        594    A
MIO                 CALL   ABL         CALL ABL(FBLK,L)                              MIO        595    A
MIO                 EQ     RBS                                                       MIO        596    A
MIO        #ECS     ENDIF                                                            MIO        597    A
MIO                                                                                  MIO        598    A
MIO        RBSA     VFD    1/1,23/0,18/8,18/RBSB                                     MIO        599    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO                 VFD    1/1,2/0,18/8,3/0,18/RBSB,18/0                          MIO          600     A
MIO       *         RBSB - DUMMY BLOCK ( BOS,EOQ )                                FCC0415       18     A
MIO                                                                              MIO          601     A
MIO        D.FT=1P  EQU    D.FTP                                                 MIO          602     A
MIO        D.FT=1L  EQU    1                                                     MIO          603     A
MIO        D.NRXP   EQU    D.NRP-1     *NR* EXPONENT BIT                         MIO          604     A
MIO        D.NRXL   EQU    1                                                     MIO          605     A
MIO                                                                              MIO          606     A
MIO        RBSB     VFD    12/1S10+OC.BOS,48/0                                   MIO          607     A
MIO                 DATA   8                                                     MIO          608     A
MIO                 BFMW   D,(TY,NRX,^D,PI,JFT)                                  MIO          609     A
MIO                 DATA   0                                                     MIO          610     A
MIO                 VFD    12/2000B,48/0                                         MIO          611     A
MIO        RBSC     BSSZ   1                                                     MIO          612     A
MIO                 BFMW   D,(TY,FT=1,NRX,^D,BM,PI,JFT)                          MIO          613     A
MIO                 DATA   0                                                     MIO          614     A
MIO        RNB      TITLE  RNB - READ NEXT BLOCK                                 MIO          615     A
MIO       **        RNB - READ NEXT BLOCK OF *FTNOPT* FILE IN A SEQUENTIAL MANNER.  MIO       616     A
MIO       *                THIS SUBROUTINE MAY BE CALLED FROM *FBV* ONLY.      MIO          617     A
MIO       *                                                                    MIO          618     A
MIO       *         ENTRY  (X2) = BIW, BLOCK INDEX WORD                         MIO          619     A
MIO       *                (X3) = FBLK, FWA OF WORKING STORAGE TO READ BLOCK TO. MIO         620     A
MIO       *                (X5) = RIW, RANDOM INDEX WORD                       MIO          621     A
MIO       *                                                                    MIO          622     A
MIO       *         CALLS  GMB                                                 MIO          623     A
MIO                                                                              MIO          624     A
MIO        RNB      ENTRY. **                                                   MIO          625         I
          -CCGNC
MIO        RNB      ENTRY. **,#                                                 CCGNC        499     A
MIO                 PRNTM  RNB,RIW,X5                                            MIO          626     A
MIO                                                                              MIO          627     A
MIO        #RM      IFEQ   CP#RM,0                                              MIO          628     A
MIO                                                                              MIO          629     A
MIO                 BX6    X5                                                    MIO          630     A
MIO                 LX6    59-RI.ATP                                             MIO          631     A
MIO                                                                              MIO          632     A
MIO                 MI     X6,RNB2     IF AT[RIW]    */ IN ECS/LCM               MIO          633     A
MIO                 SX0    X3-1                                                  MIO          634     A
MIO                 SA1    RNS                                                   MIO          635     A
MIO                 NZ     X1,RNB1     IF RNS " 0    */ FILE READ STARTED        MIO          636     A
MIO                 SX7    B1                                                    MIO          637     A
MIO                 SA7    A1          RNS = 1       */ INDICATE READ STARTED    MIO          638     A
MIO                 SA3    O.RRB                                                 MIO          639     A
MIO                 SA4    MAX.RL                                                MIO          640         I
          -CCG1
MIO                 SA7    F.OPT+6     DA = 1                                    MIO          641     A
MIO                 LX4    1                                                     MIO          642     A
MIO                 SETFET OPT,X3,X4,READ,NOADD      SET FET FOR READ             MIO          643         I
          -CCG1
MIO                 SETFET OPT,X3,CO.OBL,READ,NOADD  SETUP FET FOR READ           CCG1           9         I
          -CCGNC
MIO                 SETFET OPT,X3,HO$OBL,READ,NOADD  SETUP FET FOR READ           CCGNC        500     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO                 READNS F.OPT                                                    MIO        644    A
MIO                                                                                 MIO        645    A
MIO       RNB1      LX5    -RI.LENP                                                 MIO        646    A
MIO                 READW  F.OPT,X0,X5+2     READW( OPT , FBLK , LEN[RIW]+2 )       MIO        647    A
MIO                 ZR     X1,RNB      IF EOR = 0                                  MIO        648    A
MIO                                                                                 MIO        649    A
MIO                 LX5    RI.LENP                                                  MIO        650    A
MIO                 READNS X2          RESTART READ                                MIO        651    A
MIO                 EQ     RNB1                                                     MIO        652    A
MIO       #RM       ENDIF                                                           MIO        653    A
MIO                                                                                 MIO        654    A
MIO       RNB2      CALL   GMB         GMB( FBLK , RIW )                           MIO        655    A
MIO                 EQ     RNB                                                      MIO        656    A
MIO                                                                                 MIO        657    A
MIO       RNS       BSSZ   1           READ NS STARTED FLAG                        MIO        658    A
MIO       SMB       TITLE  SMB - SAVE MEMORY BLOCK                                 MIO        659    A
MIO      **         SMB - SAVE MEMORY BLOCK                                        MIO        660    A
MIO      *                                                                         MIO        661    A
MIO      *          ENTRY  (B2) = FLCM, OLD LCM ADDRESS                            MIO        662    A
MIO      *                 (X4) = FSCM, SCM FWA OF BLOCK                           MIO        663    A
MIO      *                 (X5) = LEN, LENGTH OF BLOCK                             MIO        664    A
MIO      *                                                                         MIO        665    A
MIO      *          EXIT   (X1) = RIW, RANDOM INDEX WORD                           MIO        666    A
MIO      *                 BLOCK WRITTEN TO RWB, LCM, OR DISK                      MIO        667    A
MIO      *                                                                         MIO        668    A
MIO      *          CALLS  SBL,SBD,ASL.                                            MIO        669    A
MIO                                                                                 MIO        670    A
MIO      *          MOVE BLOCK INTO RWB                                            MIO        671    A
MIO                                                                                 MIO        672    A
MIO       SMB0      IX3    X1+X2       FDEST = RLEN+O.RWB                          MIO        673    A
MIO                 BX2    X4                                                       MIO        674    A
MIO                 SA4    O.DISK                                                   MIO        675    A
MIO                 SA7    RLEN        RLEN = NL                                   MIO        676    A
MIO                 LX1    RI.OFSP     OFS[RIW] = OL                               MIO        677    A
MIO                 LX4    RI.RAP      RA[RIW] = O.DISK                            MIO        678    A
MIO                 BX7    X1+X4                                                    MIO        679    A
MIO                 SX1    X5          L = LEN                                     MIO        680    A
MIO                 LX5    RI.LENP     LEN[RIW] = LEN                              MIO        681    A
MIO                 BX5    X5+X7                                                    MIO        682    A
MIO                 MOVE   X1,X2,X3    MOVE(L,FSCM,FDEST)                          MIO        683    A
MIO                 BX1    X5                                                       MIO        684    A
MIO                                                                                 MIO        685    A
MIO       SMB1      SX2    B1+B1                                                    MIO        686    A
MIO                 LX2    RI.LENP                                                  MIO        687    A
MIO                 IX1    X1-X2       LEN[RIW] = LEN[RIW] - 2                     MIO        688    A
MIO                                                                                 MIO        689    A
MIO                 PRNTM  SMB,RIW,X1                                               MIO        690    A
MIO                                                                                 MIO        691    A
MIO       SMB       ENTRY. **                                                       MIO        692          I
         -CCGNC
MIO       SMB       ENTRY. **,#                                                     CCGNC      501    A
MIO                                                                                 MIO        693    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO      *         ALLOCATE SPACE IN LCM IF POSSIBLE                                MIO        694    A
MIO                                                                                MIO        695    A
MIO               SX4    X4-1        FSCM = FSCM - 1                               MIO        696    A
MIO               SX5    X5+2        LEN = LEN + 2                                 MIO        697    A
MIO               SB5    X5                                                        MIO        698    A
MIO                                                                                MIO        699    A
MIO       #ECS    IFNE   CT.ECS,0                                                  MIO        700    A
MIO               SA2    ADT                                                       MIO        701    A
MIO               ZR     X2,SMB2     IF ADT = 0    */WRITE IS TO DISK              MIO        702    A
MIO                                                                                MIO        703    A
MIO               LX5    30                                                        MIO        704    A
MIO               BX5    X4+X5                                                     MIO        705    A
MIO               CALL   ASL         FLCM = ASL(FLCM,LEN,L)  */ALLOCATE LCM        MIO        706    A
MIO               SX4    X5                                                        MIO        707    A
MIO               AX5    30                                                        MIO        708    A
MIO               PRINT  SMB,(* --FLCM FROM ASL = *,Z8),X2                         MIO        709    A
MIO                                                                                MIO        710    A
MIO               MI     X2,SMB2     IF FLCM.LT.0  */WRITE IS TO DISK              MIO        711    A
MIO                                                                                MIO        712    A
MIO      *         WRITE BLOCK TO LCM                                              MIO        713    A
MIO                                                                                MIO        714    A
MIO               SA1    X4                                                        MIO        715    A
MIO               SA3    SMBB        M1 = MASK [BH](BI,BST,HB)                     MIO        716    A
MIO               SB3    X7-1                                                      MIO        717    A
MIO               BX1    X1*X3       BHW = M1 & BHW                                MIO        718    A
MIO               SB4    X2+B3                                                     MIO        719    A
MIO               LX7    BH.LENP     LEN[BHW] = L                                  MIO        720    A
MIO               BX6    X1+X7                                                     MIO        721    A
MIO               SA6    A1          [FSCM] = BHW                                  MIO        722    A
MIO               PRNTM  SMB,BHW,X6                                                MIO        723    A
MIO               CALL   SBL         CALL SBL(FLCM,FSCM,LEN,RIW)                   MIO        724    A
MIO               MX6    -RI.LENL                                                  MIO        725    A
MIO               SX0    B4                                                        MIO        726    A
MIO               LX6    RI.LENP                                                   MIO        727    A
MIO               PRNTM  SMB,BTW,X7                                                MIO        728    A
MIO               WX7    X0,SMBA     LEN[BTW] = L                                  MIO        729    A
MIO               BX7    X1*X6                                                     MIO        730    A
MIO               LX5    RI.LENP     LEN[RIW] = LEN                                MIO        731    A
MIO               BX1    X5+X7                                                     MIO        732    A
MIO               EQ     SMB1                                                      MIO        733    A
MIO       #ECS    ENDIF                                                            MIO        734    A
MIO                                                                                MIO        735    A
MIO      *         WRITE BLOCK TO RWB OR DISK                                      MIO        736    A
MIO                                                                                MIO        737    A
MIO       SMB2    SA3    MAX.RL                                                    MIO        738         I
         -CCG1
MIO       SMB2    SX3    CO.OBL                                                    CCG1        10         I
         -CCGNC
MIO       SMB2    SX3    HO$OBL                                                    CCGNC      502    A
MIO               SA1    RLEN        OL = RLEN                                     MIO        739    A
MIO               SA2    O.RWB                                                     MIO        740    A
MIO               IX0    X5-X3                                                     MIO        741    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO                IX7    X1+X5       NL = OL + LEN                          MIO        742     A
MIO                IX6    X7-X3                                              MIO        743     A
MIO                MI     X6,SMB0     IF NL .LT. MAX.RL                      MIO        744     A
MIO                                                                           MIO        745     A
MIO      *         DUMP RWB TO DISK                                           MIO        746     A
MIO                                                                           MIO        747     A
MIO                ZR     X1,SMB3     IF OL = 0                              MIO        748     A
MIO                LX5    30                                                 MIO        749     A
MIO                BX6    X4+X5                                              MIO        750     A
MIO                SA6    SMBA                                               MIO        751     A
MIO                SA4    O.RWB       FRWB = O.RWB                           MIO        752     A
MIO                BX5    X1          L = RLEN                               MIO        753     A
MIO                CALL   SBD         CALL SBD(FRWB,L,RIW)                   MIO        754     A
MIO                SA5    SMBA                                               MIO        755     A
MIO                SA2    O.RWB                                              MIO        756     A
MIO                SX4    X5                                                 MIO        757     A
MIO                AX5    30                                                 MIO        758     A
MIO                SX7    X5                                                 MIO        759     A
MIO                MX1    0           OL = 0                                 MIO        760     A
MIO                MI     X0,SMB0     IF LEN < MAX.RL                        MIO        761     A
MIO                MX6    0                                                  MIO        762     A
MIO                SA6    RLEN        RLEN = 0                               MIO        763     A
MIO                                                                           MIO        764     A
MIO      *         WRITE BLOCK DIRECTLY FROM BLK                              MIO        765     A
MIO                                                                           MIO        766     A
MIO       SMB3     CALL   SBD         CALL SBD(FSCM,LEN,RIW)                 MIO        767     A
MIO                EQ     SMB1                                               MIO        768     A
MIO                                                                           MIO        769     A
MIO                                                                           MIO        770     A
MIO       SMBA     BSS    1                                                  MIO        771     A
MIO       SMBB     BFMW   BH,(BI,BST,HB)                                     MIO        772     A
MIO       GMB      TITLE  GMB - GET MEMORY BLOCK                             MIO        773     A
MIO      **        GMB - GET MEMORY BLOCK                                    MIO        774     A
MIO      *                                                                   MIO        775     A
MIO      *         ENTRY  (X3) = FSCM, SCM FWA OF BLOCK                      MIO        776     A
MIO      *                (X5) = RIW, RANDOM INDEX WORD                      MIO        777     A
MIO      *                                                                   MIO        778         I
         -FCC0415
MIO      *         EXIT   BLOCK TRANSFERRED                                  MIO        779         I
         -FCC0415
MIO      *                                                                   MIO        780     A
MIO      *         CALLS  GBL,GBD,MVE=.                                      MIO        781     A
MIO                                                                           MIO        782     A
MIO       GMB      ENTRY. **                                                 MIO        783         I
         -CCGNC
MIO       GMB      ENTRY. **,#                                               CCGNC      503     A
MIO                                                                           MIO        784         I
         -FCC0415
MIO                BX2    X5                                                 MIO        785     A
MIO                LX2    59-RI.ICP                                          MIO        786     A
MIO                PL     X2,GMB0     IF ^ IC[RIW]                           MIO        787     A
MIO                                                                           MIO        788     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO              LX2    1+RI.ICP-RI.LENP                                  MIO        789    A
MIO              SX1    X2+2        L = LEN[RIW] + 2                      MIO        790    A
MIO              LX2    RI.LENP-RI.FWAP                                   MIO        791    A
MIO              SX2    X2-1        FBLK = FWA[RIW] - 1                   MIO        792    A
MIO              SX3    X3-1        FSCM = FSCM - 1                       MIO        793    A
MIO              MOVE   X1,X2,X3    MOVE(L,FBLK,FSCM)                     MIO        794    A
MIO              EQ     GMB                                               MIO        795    A
MIO                                                                      MIO        796    A
MIO       GMB0   SA4    MAX.RL                                            MIO        797         I
         -CCG1
MIO              SB4    X4                                                MIO        798         I
         -CCG1
MIO       GMB0   SB4    CO.OBL                                            CCG1        11         I
         -CCGNC
MIO       GMB0   SB4    HO$OBL                                            CCGNC      504    A
MIO              PRNTM  GMB,RIW,X5                                        MIO        799    A
MIO                                                                      MIO        800    A
MIO       #ECS   IFNE   CT.ECS,0                                          MIO        801    A
MIO              BX6    X5                                                MIO        802    A
MIO              LX6    59-RI.ATP                                         MIO        803    A
MIO              PL     X6,GMB1     IF ^AT[RIW]   */READ FROM DISK        MIO        804    A
MIO                                                                      MIO        805    A
MIO      *       READ BLOCK FROM LCM                                     MIO        806    A
MIO                                                                      MIO        807    A
MIO              LX5    -RI.LCMP                                          MIO        808    A
MIO              SX4    X5          FLCM = LCM[RIW]                       MIO        809    A
MIO              LX5    RI.LCMP-RI.LENP                                   MIO        810    A
MIO              SX5    X5+2        L = LEN[RIW]+2                        MIO        811    A
MIO              SX2    X3-1                                              MIO        812    A
MIO              CALL   GBL         CALL GBL(FLCM,L,FSCM)                 MIO        813    A
MIO              SA2    X2          BHW = [FSCM]                          MIO        814    A
MIO              PRNTM  GMB,BHW,X2                                        MIO        815    A
MIO              SA4    BSI                                               MIO        816    A
MIO              MX0    -BH.LENL    M1 = MASK[BH](LEN)                    MIO        817    A
MIO              LX2    -BH.LENP                                          MIO        818    A
MIO              BX6    X0*X2       BHW = ^M1 & BHW                       MIO        819    A
MIO              LX4    BH.BSTP     BST[BHW] = BSI                        MIO        820    A
MIO              SX0    B1                                                MIO        821    A
MIO              LX0    BH.PRIP     PRI[BHW] = 1                          MIO        822    A
MIO              BX7    X5+X6       LEN[BHW] = L                          MIO        823    A
MIO              IX4    X0+X4                                             MIO        824    A
MIO              BX7    X4+X7                                             MIO        825    A
MIO              SA7    A2          [FSCM] = BHW                          MIO        826    A
MIO              PRNTM  GMB,BHW,X7                                        MIO        827    A
MIO              EQ     GMB                                               MIO        828    A
MIO       #ECS   ENDIF                                                    MIO        829    A
MIO                                                                      MIO        830    A
MIO      *       CHECK IF BLOCK IS IN RRB OR RWB                         MIO        831    A
MIO                                                                      MIO        832    A
MIO       GMB1   LX5    -RI.RAP                                           MIO        833    A
MIO              MX0    -RI.RAL                                           MIO        834    A
MIO              BX4    -X0*X5      RA = RA[RIW]                          MIO        835    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
      MIO                SA1    O.RRB                                                  MIO          836     A
      MIO                SA2    N.RRB                                                  MIO          837     A
      MIO                LX5    RI.RAP-RI.LENP                                         MIO          838     A
      MIO                PRINT  GMB,(*  --N.RRB =*Z7),N.RRB                            MIO          839     A
      MIO                SB2    X1          BUF = [O.RRB]                              MIO          840     A
      MIO                IX6    X2-X4                                                  MIO          841     A
      MIO                ZR     X6,GMB2     IF N.RRB = RA */BLOCK IN RRB               MIO          842     A
      MIO                                                                              MIO          843     A
      MIO                SA1    O.RWB                                                  MIO          844     A
      MIO                SA2    O.DISK                                                 MIO          845     A
      MIO                PRINT  GMB,(* --O.DISK =*Z9),O.DISK                           MIO          846     A
      MIO                SB2    X1          BUF = [O.RWB]                              MIO          847     A
      MIO                IX6    X2-X4                                                  MIO          848     A
      MIO                ZR     X6,GMB2     IF O.DISK = RA   */BLOCK IN RWB            MIO          849     A
      MIO                                                                              MIO          850     A
      MIO                SB5    X5                                                     MIO          851         I
               -FCC1352
      MIO                GT     B5,B4,GMB3  IF L .GT. MAX.RL */READ TO *BLK*           MIO          852         I
               -FCC1352
      MIO                SB5    X5+2                                                   FCC1352        5     A
      MIO                GE     B5,B4,GMB3  IF LEN[RIW]+2 \ MAX.RL                     FCC1352        6         I
               -CCG1
      MIO                GE     B5,B4,GMB3  IF LEN[RIW]+2 \ CO.OBL                     CCG1          12         I
               -CCGNC
      MIO                GE     B5,B4,GMB3  IF LEN[RIW]+2 \ HO$OBL                     CCGNC        505     A
      MIO                                                                              MIO          853     A
      MIO                BX6    X3                                                     MIO          854     A
      MIO                BX7    X5                                                     MIO          855     A
      MIO                SA2    O.RRB                                                  MIO          856     A
      MIO                SA6    GMBA                                                   MIO          857     A
      MIO                SA7    A6+B1                                                  MIO          858     A
      MIO                BX7    X4                                                     MIO          859     A
      MIO                SX5    B4          L = MAX.RL                                 MIO          860         I
               -CCG1
      MIO                SX5    B4          L= CO.OBL                                  CCG1          13         I
               -CCGNC
      MIO                SX5    B4          L= HO$OBL                                  CCGNC        506     A
      MIO                SA7    N.RRB                                                  MIO          861     A
      MIO                CALL   GBD         CALL GBD(O.RRB,L,RA)                       MIO          862     A
      MIO                SA1    O.RRB                                                  MIO          863     A
      MIO                SA3    GMBA                                                   MIO          864     A
      MIO                SA5    A3+B1                                                  MIO          865     A
      MIO                SB2    X1                                                     MIO          866     A
      MIO                                                                              MIO          867     A
      MIO       GMB2     SX1    X5+2        L = LEN[RIW]+2                             MIO          868     A
      MIO                LX5    RI.LENP-RI.OFSP                                        MIO          869     A
      MIO                MX0    -RI.OFSL                                               MIO          870     A
      MIO                BX7    -X0*X5      OFS = OFS[RIW]                             MIO          871     A
      MIO                SX2    X7+B2       FBLK = BUF+OFS                             MIO          872     A
      MIO                SX3    X3-1        FSCM = FSCM-1                              MIO          873     A
      MIO                SA0    X3                                                     MIO          874     A
      MIO                MOVE   X1,X2,X3    MOVE(L,FBLK,FSCM)                          MIO          875     A

               0         1         2         3         4         5         6         7         8
               12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO                EQ     GMB4                                                  MIO          876     A
MIO                                                                             MIO          877     A
MIO       *        READ DIRECTLY INTO BLK                                        MIO          878     A
MIO                                                                             MIO          879     A
MIO        GMB3    SX2    X3-1        FSCM = FSCM-1                              MIO          880     A
MIO                SA0    X2                                                     MIO          881     A
MIO                SX5    X5+2        L = LEN[RIW]+2                             MIO          882     A
MIO                CALL   GBD         CALL GBD(FSCM,L,RA)                        MIO          883     A
MIO                                                                             MIO          884     A
MIO        GMB4    SA2    A0                                                     MIO          885     A
MIO                SA4    BSI                                                    MIO          886     A
MIO                SX0    B1                                                     MIO          887     A
MIO                LX0    BH.PRIP     PRI[BHW] = 1                               MIO          888     A
MIO                LX4    BH.BSTP                                                MIO          889     A
MIO                IX4    X0+X4                                                  MIO          890     A
MIO                BX6    X2+X4                                                  MIO          891     A
MIO                SA6    A2                                                     MIO          892     A
MIO                PRNTM  GMB,BHW,X6                                             MIO          893     A
MIO                EQ     GMB                                                    MIO          894     A
MIO                                                                             MIO          895     A
MIO        GMBA    BSS    2                                                      MIO          896     A
MIO                TTL    MIO - MASS STORAGE RANDOM I/O (INTERNAL ROUTINES)      MIO          897     A
MIO        DMB     TITLE  DMB - DUMP MEMORY BLOCKS                               MIO          898     A
MIO       **       DMB - DUMP MEMORY BLOCKS                                      MIO          899     A
MIO       *                                                                     MIO          900     A
MIO       *        ENTRY  (X1) = LEN, LENGTH OF SPACE REQUIRED                   MIO          901     A
MIO       *               (B2) = 0, GET ALL REQ.SPACE BY DUMPING BASIC BLOCKS    RSM2763X      66     A
MIO       *                    " 0, DUMP INACTIVE BLOCKS ONLY. IF NOT ENOUGH     RSM2763X      67     A
MIO       *                         SPACE OBTAINED,CALL MOREFL TO EXTEND FL      RSM2763X      68     A
MIO       *                                                                     MIO          902     A
MIO       *        EXIT   (X1) = 0, FREE SPACE WAS OBTAINED                      MIO          903         I
          -RSM2763X
MIO       *        EXIT   (X1) = 0 - REQUIRED SPACE OBTAINED                     RSM2763X      69     A
MIO       *               (X1) " 0 - NOT ENOUGH SPACE OBTAINED,X1 = NEW WN       RSM2763X      70     A
MIO       *               (X2) = WFL - WORDS OBTAINED BY CALLING MOREFL          RSM2763X      71     A
MIO       *                    " 0, ERROR                                        MIO          904         I
          -RSM2763X
MIO       *                                                                     MIO          905     A
MIO       *        CALLS  SMB,ABB.                                               MIO          906     A
MIO                                                                             MIO          907     A
MIO        DMB     ENTRY. **                                                     MIO          908         I
          -CCGNC
MIO        DMB     ENTRY. **,#                                                   CCGNC        507     A
MIO                SA2    =XOPT2                                                 MIO          909         I
          -FCC0415
MIO                SB7    X1                                                     MIO          910     A
MIO                SX6    B2                                                     RSM2763X      72     A
MIO                SA6    DMBH        SAVE B2                                    RSM2763X      73     A
MIO                MX1    0                                                      MIO          911         I
          -FCC0415
MIO                ZR     X2,DMB      IF OPT2 = 0   */ OPT=1                     MIO          912         I
          -FCC0415

          0        1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO                BX6    X0                                                  MIO          913     A
MIO                BX7    X5                                                  MIO          914     A
MIO                SA6    DMBG        SAVE X0, X5                             MIO          915     A
MIO                SA7    DMBG+1                                              MIO          916     A
MIO                SX6    A0                                                  MIO          917     A
MIO                SX7    A5                                                  MIO          918     A
MIO                SA6    DMBG+2      SAVE A0,A5                              MIO          919     A
MIO                SA7    DMBG+3                                              MIO          920     A
MIO                                                                           MIO          921     A
MIO                PRINT  DMB,(* -- REQUEST FOR*Z7* WORDS*),B7                 MIO          922     A
MIO                SA5    DMBA        M1 = MASK FOR AV AND FW FLAGS           RSM2763X      74     A
MIO       DMB0     SA2    O.BST                                               RSM2763X      75     A
MIO                SA2    O.BST                                               MIO          923          I
          -RSM2763X
MIO                SA3    O.SEQ                                               MIO          924     A
MIO                SA4    O.BLK       FBLK = O.BLK                            MIO          925     A
MIO                SA5    DMBA        M1 = MASK [BH](FW)                      MIO          926          I
          -RSM2763X
MIO                SB6    60-BH.LENP                                          MIO          927     A
MIO                SB3    X2                                                  MIO          928     A
MIO                SB4    X3-1        SEQ = O.SEQ - 1                         MIO          929     A
MIO                SA1    O.BIT                                               MIO          930     A
MIO                SA3    A5+B1                                               MIO          931     A
MIO                SB2    X1+B1                                               MIO          932     A
MIO                BX0    X3                                                  MIO          933     A
MIO                SA4    X4+B1       BHW = [FBLK+1]                          MIO          934     A
MIO                PRNTM  DMB,BHW,X4                                          MIO          935          I
          -RSM2763X
MIO                                                                           MIO          936     A
MIO       *        LOOP TO FIND BLOCK TO DUMP                                MIO          937     A
MIO                                                                           MIO          938     A
MIO       DMB1     ZR     X4,DMB3     IF BHW = 0    */END OF TABLE            MIO          939     A
MIO                                                                           MIO          940     A
MIO                BX6    X5*X4       TEST = M1 & BHW                         MIO          941     A
MIO                SB5    A4          OBLK = FBLK                             MIO          942     A
MIO                PRNTM  DMB,BHW,X4                                          RSM2763X      76     A
MIO                LX2    B6,X4       TBHW = BHW                              MIO          943     A
MIO                BX6    X0-X6       TEST = TEST .XOR. M3                    MIO          944     A
MIO                SA4    X2+B5       BHW = [FBLK+LEN[BHW]]                   MIO          945     A
MIO                PRNTM  DMB,BHW,X4                                          MIO          946          I
          -RSM2763X
MIO                NZ     X6,DMB1     IF TEST " 0                             MIO          947     A
MIO                                                                           MIO          948     A
MIO                EQ     B4,B5,DMB1  OBLK = SEQ    */ DO NOT DUMP ACTIVE BLOCK MIO          949     A
MIO                                                                           MIO          950     A
MIO       *        DUMP BLOCK TO MASS STORAGE                                MIO          951     A
MIO                                                                           MIO          952     A
MIO                PRINT  DMB,(* ABOVE BLOCK DUMPED TO MS. RIW FOLLOWS *)     RSM2763X      77     A
MIO                MX6    -BH.BSTL                                            MIO          953     A
MIO                LX6    BH.BSTP                                             MIO          954     A
MIO                MX3    -BA.SVLL                                            MIO          955     A
MIO                SX5    X2          L = LEN[BHW]                            MIO          956     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO              BX6    X2*X6                                                  MIO          957     A
MIO              SA6    B5                                                     MIO          958     A
MIO              LX2    BH.LENP-BH.BSTP   B = BST[BHW]                         MIO          959     A
MIO              SB4    X2                                                     MIO          960     A
MIO              MI     B4,DMB1A    IF B < 0      */NO BST ENTRY               MIO          961     A
MIO                                                                            MIO          962     A
MIO              SA1    X2+B3       BAW = BST(B)                               MIO          963     A
MIO              LX3    BA.SVLP                                                MIO          964     A
MIO              BX7    -X3*X1                                                 MIO          965     A
MIO              PRNTM  DMB,BAW,X7                                             MIO          966     A
MIO              SA7    A1          BST(B) = BAW                               MIO          967     A
MIO                                                                            MIO          968     A
MIO       DMB1A  LX2    BH.BSTP-BH.BIP    AR = BI[BHW]                         MIO          969     A
MIO              SA3    X2+B2       RIW = BIT(AR+1)                            MIO          970     A
MIO              LX3    59-RI.ATP                                              MIO          971     A
MIO              SX7    B7                                                     MIO          972         I
          -FCC1123
MIO              SB2    X4          L=LEN[BHW]                                 FCC1123        4     A
MIO              NE     B1,B2,DMB1B IF L = 1       */SKIP 1 WORD FREE SPACE    FCC1123        5     A
MIO              SA4    A4+1                                                   FCC1123        6     A
MIO       DMB1B  SX7    B7                                                     FCC1123        7     A
MIO              IX6    X7-X5       LEN = LEN - L                              MIO          973     A
MIO              SX0    A5                                                     MIO          974     A
MIO              LX6    18                                                     MIO          975     A
MIO              SX1    A4                                                     MIO          976     A
MIO              BX6    X0+X6                                                  MIO          977     A
MIO              LX6    18                                                     MIO          978     A
MIO              BX6    X1+X6                                                  MIO          979     A
MIO              SA6    DMBE        T1 = LEN, M1, FBLK                         MIO          980     A
MIO              SX7    B5                                                     MIO          981     A
MIO              SX0    A3                                                     MIO          982     A
MIO              LX7    18                                                     MIO          983     A
MIO              BX7    X0+X7                                                  MIO          984     A
MIO              SX1    X5                                                     MIO          985     A
MIO              LX7    18                                                     MIO          986     A
MIO              BX7    X1+X7                                                  MIO          987     A
MIO              SA7    DMBF        T2 = OBLK, AR, L                           MIO          988     A
MIO              SX5    X5-2                                                   MIO          989     A
MIO              SB2    B0          FLCM = 0                                   MIO          990     A
MIO              PL     X3,DMB2     IF AT[RIW] = 0                             MIO          991     A
MIO                                                                            MIO          992     A
MIO              LX3    1+RI.ATP-RI.LCMP                                       MIO          993     A
MIO              SB2    X3          FLCM = LCM[RIW]                            MIO          994     A
MIO                                                                            MIO          995     A
MIO       DMB2   SX4    B5+B1       FSCM = FBLK + 1                            MIO          996     A
MIO              CALL   SMB         CALL SMB(FLCM,L,FBLK) */SAVE BLOCK ON MS   MIO          997     A
MIO              SA2    DMBF                                                   MIO          998     A
MIO              SB3    X2                                                     MIO          999     A
MIO              BX7    X1                                                     MIO         1000     A
MIO              AX2    18                                                     MIO         1001     A
MIO              SA7    X2          BIT(AR+1) = RIW                            MIO         1002     A
MIO              PRNTM  DMB,RIW,X7                                             MIO         1003     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO              AX2    18                                                  MIO          1004     A
MIO              SB2    X2                                                  MIO          1005     A
MIO              CALL   ABB         CALL ABB(OBLK,L)                        MIO          1006     A
MIO              SA1    DMBE                                                MIO          1007     A
MIO              SA4    X1                                                  MIO          1008     A
MIO              AX1    18                                                  MIO          1009     A
MIO              SA5    X1                                                  MIO          1010     A
MIO              AX1    18                                                  MIO          1011     A
MIO              SB7    X1                                                  MIO          1012     A
MIO              SA3    A5+1                                                MIO          1013     A
MIO              LE     B7,B0,DMB4  IF LEN @ 0                             MIO          1014     A
MIO                                                                        MIO          1015     A
MIO              BX0    X3                                                  MIO          1016     A
MIO              SA1    O.BIT                                               MIO          1017     A
MIO              SA2    O.BST                                               MIO          1018     A
MIO              SA3    O.SEQ                                               MIO          1019     A
MIO              SB6    60-BH.LENP                                          MIO          1020     A
MIO              SB2    X1+B1                                               MIO          1021     A
MIO              SB3    X2                                                  MIO          1022     A
MIO              SB4    X3-1                                                MIO          1023     A
MIO              EQ     DMB1                                                MIO          1024     A
MIO                                                                        MIO          1025     A
MIO      *       GET MASK FOR NEXT GROUP                                    MIO          1026     A
MIO                                                                        MIO          1027     A
MIO       DMB3   SA5    A5+2        */ M3 OR M5 MASK OR ZERO               RSM2763X       78     A
MIO              ZR     X5,DMB4     IF (X5) = 0 */ LAST BLK SCAN DONE      RSM2763X       79     A
MIO              SA3    A5+B1                                              RSM2763X       80     A
MIO              BX0    X3          */ M4 OR M6 MASK FOR 2ND OR 3RD SCAN   RSM2763X       81     A
MIO              ZR     X0,DMB3A    IF M4 MASK */ 2ND SCAN IS TO START     RSM2763X       82     A
MIO              SA4    DMBH                                               RSM2763X       83     A
MIO              ZR     X4,DMB3A    IF  DMBH  = 0 */ STANDARD PROCESSING   RSM2763X       84     A
MIO                                                                        RSM2763X       85     A
MIO      *       SPECIAL PROCESSING REQUESTED - CALL MOREFL TO GET          RSM2763X       86     A
MIO      *       REMAINING NO. OF WORDS BY EXTENDING FL RATHER THEN BY      RSM2763X       87     A
MIO      *       DUMPING THE HIGHEST PRIORITY BLOCKS                        RSM2763X       88     A
MIO                                                                        RSM2763X       89     A
MIO              SX1    B7          WN = NO.OF WORDS NEEDED                RSM2763X       90     A
MIO              PRINT  DMB,(*  DMB CALLS MOREFL,WN =*Z7),(X1)             RSM2763X       91     A
MIO              CALL   MOREFL      MOREFL(X6) = NO.OF WORDS GRANTED       RSM2763X       92     A
MIO              SA6    DMBI        SAVE WG - NO.OF WORDS GRANTED          RSM2763X       93     A
MIO              IX6    X1-X6                                              RSM2763X       94     A
MIO              SB7    X6          RWN = REM.NO.OF WORDS NEEDED           RSM2763X       95     A
MIO              LE     B7,DMB4     IF RWN @ 0                             RSM2763X       96     A
MIO              SA5    DMBA+4      M5 = MASK FOR 3RD SCAN                 RSM2763X       97     A
MIO              EQ     DMB0        */ GET RWN BY DUMPING THE TOP PRIOR.BLOCKS RSM2763X   98     A
MIO                                                                        RSM2763X       99     A
MIO       DMB3A  SA4    O.BLK                                              RSM2763X      100     A
MIO              SA4    X4+B1                                              RSM2763X      101     A
MIO              EQ     DMB1                                               RSM2763X      102     A
MIO       DMB3   SA5    A5+2                                               MIO          1028          I
         -RSM2763X
MIO              SA4    O.BLK                                              MIO          1029          I

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

          -RSM2763X
MIO                 SA3    A5+B1                                                MIO        1030        I
          -RSM2763X
MIO                 ZR     X5,DMB4     IF M1 = 0                                MIO        1031        I
          -RSM2763X
MIO                                                                             MIO        1032        I
          -RSM2763X
MIO                 SA4    X4+B1                                                MIO        1033        I
          -RSM2763X
MIO                 BX0    X3                                                   MIO        1034        I
          -RSM2763X
MIO                 EQ     DMB1                                                 MIO        1035        I
          -RSM2763X
MIO                                                                             MIO        1036    A
MIO        DMB4     SX1    B7                                                   MIO        1037    A
MIO                 SA5    DMBG+3                                               MIO        1038    A
MIO                 SA4    DMBG+2                                               MIO        1039    A
MIO                 SA3    DMBG+1                                               MIO        1040    A
MIO                 SA2    DMBG                                                 MIO        1041    A
MIO                 SA0    X4          RESTORE A0, A5                           MIO        1042    A
MIO                 SA5    X5                                                   MIO        1043    A
MIO                 BX5    X3                                                   MIO        1044    A
MIO                 BX0    X2          RESTORE X0,X5                            MIO        1045    A
MIO                 PRINT  DMB,(* -- SUMMARY -- B7 =*Z7),B7                     MIO        1046    A
MIO                 PRNTM  DMB,BIT                                              MIO        1047    A
MIO                 PRNTM  DMB,BST                                              MIO        1048    A
MIO                 PRNTM  BLK,,=1                                              MIO        1049    A
MIO                                                                             MIO        1050    A
MIO                 SA2    DMBI        WFL = NO OF WORDS GRANTED BY MOREFL      RSM2763X    103    A
MIO                 MX6    0                                                    RSM2763X    104    A
MIO                 SA6    A2          DMBI = 0                                 RSM2763X    105    A
MIO                 PL     B7,DMB      IF NOT ENOUGH SPACE FREED                MIO        1051    A
MIO                 MX1    0           X1 = 0        */ SUCCESS                 MIO        1052    A
MIO                 EQ     DMB                                                  MIO        1053    A
MIO                                                                             MIO        1054    A
MIO        DMBA     BFMW   BH,(AV,FW)                                           MIO        1055    A
MIO                 BFMW   BH,(FW)                                              MIO        1056    A
MIO                 BFMW   BH,(AV,PRI)                                          MIO        1057    A
MIO                 BSSZ   1                                                    MIO        1058    A
MIO                 BFMW   BH,(AV,PRI)                                          MIO        1059    A
MIO                 BFMW   BH,(PRI)                                             MIO        1060    A
MIO                 BSSZ   1                                                    MIO        1061    A
MIO        DMBE     BSS    1                                                    MIO        1062    A
MIO        DMBF     BSS    1                                                    MIO        1063    A
MIO        DMBG     BSS    4                                                    MIO        1064    A
MIO        DMBH     BSS    1                                                    RSM2763X    106    A
MIO        DMBI     BSSZ   1                                                    RSM2763X    107    A
MIO        ASB      TITLE  ASB - ALLOCATE SPACE IN BLK                          MIO        1065    A
MIO       **        ASB - ALLOCATE SPACE IN BLK                                 MIO        1066    A
MIO       *                                                                     MIO        1067    A
MIO       *         ENTRY  (B2) = ADDRESS OF BLOCK HEADER WORD                  MIO        1068    A
MIO       *                (B5) = OL, CURRENT LENGTH OF BLOCK                   MIO        1069    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO      *                (X1) = NL, NEW LENGTH OF BLOCK                          MIO         1070     A
MIO      *                                                                        MIO         1071     A
MIO      *         EXIT   (X3) = FWA, FWA OF BLOCK ALLOCATED                       MIO         1072     A
MIO                                                                               MIO         1073     A
MIO       ASB0     SA1    =7LMIO-ASB        TABLE OVERFLOW                         MIO         1074          I
         -FCC0415
MIO                CALL   PUNT                                                    MIO         1075          I
         -FCC0415
MIO                                                                               MIO         1076          I
         -FCC0415
MIO       ASB      ROUTINE                                                        MIO         1077     A
MIO                SB7    B1+B1                                                   MIO         1078     A
MIO                SB6    RBSB-1                                                  MIO         1079     A
MIO                PRINT  ASB,(* BLK SIZE=*Z7),(L.BLK)                            RSM2763X     108     A
MIO                SB3    X1+B7       NL = NL + 2                                 MIO         1080     A
MIO                EQ     B5,B7,ASB1  IF OL = 2     */NO SPACE ALLOCATED          MIO         1081     A
MIO                                                                               MIO         1082     A
MIO                EQ     B2,B6,ASB1  IF FBLK = RBSB  */DUMMY BLOCK              MIO         1083     A
MIO                                                                               MIO         1084     A
MIO                EQ     B5,B3,ASB5  IF OL = NL                                  MIO         1085     A
MIO                                                                               MIO         1086     A
MIO                GT     B5,B3,ASB4  IF OL > NL                                  MIO         1087     A
MIO                                                                               MIO         1088     A
MIO                SX4    B3                                                      MIO         1089     A
MIO                SB3    B5                                                      MIO         1090     A
MIO                CALL   ABB         CALL ABB(FBLK,OL)                          MIO         1091     A
MIO                SB3    X4+                                                     MIO         1092     A
MIO                                                                               MIO         1093     A
MIO      *         DUMP BLOCKS FROM BLK IF NEED EXTRA SPACE.                       MIO         1094     A
MIO                                                                               MIO         1095     A
MIO       ASB1     SA1    MX.AVS      MAX = MX.AVS                                MIO         1096     A
MIO                SB6    X1+                                                     MIO         1097     A
MIO                SX2    X1-100B                                                 MIO         1098     A
MIO                PRINT ASB,(* MX.AVS,NL=*2Z7),(B6,B3)                           RSM2763X     109     A
MIO                MI     X2,ASB1C                                                MIO         1099     A
MIO                                                                               MIO         1100     A
MIO                GE     B6,B3,ASB1A IF MAX \ NL                                 MIO         1101     A
MIO                                                                               MIO         1102     A
MIO       ASB1C    SX1    B3-B6       DL = NL - MAX.AS                            MIO         1103     A
MIO                SA3    =XOPT2                                                  FCC0415       19          I
         -CCG7
MIO                SA3    =XCO.OPT                                                CCG7          91          I
         -CCGNC
MIO                SA3    =XHO$OPT                                                CCGNC        508     A
MIO                SX6    B3                                                      MIO         1104     A
MIO                SA6    ASBA                                                    MIO         1105     A
MIO                PRINT  ASB,(* SPACE NEEDED = *,Z8),B3                          MIO         1106     A
MIO                ZR     X3,ASB1E    IF OPT2 = 0    */OPT = 1                    FCC0415       20          I
         -CCG7
MIO                PL     X3,ASB1E    IF OPT " 2                                  CCG7          92     A
MIO                SX2    401B                                                    MIO         1107     A
MIO                IX7    X1-X2                                                   MIO         1108     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO                PL     X7,ASB1B    DL = MAX( DL , 401B )                          MIO         1109     A
MIO                BX1    X2                                                         MIO         1110     A
MIO       ASB1B    SX6    B0                                                         RSM2763X     110     A
MIO                SB2    B1          */ SPEC. PROCESSING BY DMB                     RSM2763X     111     A
MIO       ASB1B    SX6    0                                                          MIO         1111          I
         -RSM2763X
MIO                PRINT  ASB,(* ASB CALLS DMB,WN=*Z7),(X1)                          RSM2763X     112     A
MIO                SA6    O.SEQ       O.SEQ = 0                                      MIO         1112     A
MIO                SA6    L.SEQ       L.SEQ = 0                                      MIO         1113     A
MIO                CALL   DMB         DUMP BLOCKS TO FREE UP SPACE                   MIO         1114     A
MIO                NZ     X1,ASB6     IF NOT ENOUGH SPACE OBTAINED                   RSM2763X     113     A
MIO                ZR     X2,ASB1D    ENOUGH SPACE BY DUMPING BLOCKS ONLY            RSM2763X     114     A
MIO                                                                                 RSM2763X     115     A
MIO      *         DMB HAD TO CALL MOREFL TO GET ENOUGH SPACE, X2 = WFL             RSM2763X     116     A
MIO                                                                                 RSM2763X     117     A
MIO                BX1    X2                                                         RSM2763X     118     A
MIO                NZ     X1,ASB0     IF NOT ENOUGH SPACE AVAILABLE                  MIO         1115          I
         -FCC0415
MIO                SA2    ASBA                                                       MIO         1116          I
         -FCC0415
MIO                ZR     X1,ASB1D    IF ENOUGH SPACE OBTAINED                       FCC0415       21          I
         -RSM2763X
MIO                                                                                 FCC0415       22     A
MIO      *         INCREASE SIZE OF *BLK*                                           FCC0415       23     A
MIO                                                                                 FCC0415       24     A
MIO       ASB1E    SA2    =XTO        DMBH = [TO]    */SAVE TABLE OVERFLOW EXIT      FCC0415       25          I
         -RSM2763X
MIO                SX6    =XTOV                                                      FCC0415       26          I
         -RSM2763X
MIO                BX7    X2                                                         FCC0415       27          I
         -RSM2763X
MIO                SA6    A2          [TO] = TOV     */SET TABLE OVERFLOW EXIT       FCC0415       28          I
         -RSM2763X
MIO                SA7    ASBB                                                       FCC0415       29          I
         -RSM2763X
MIO                SX1    X1+100B     LEN = LEN + 100B                               FCC0415       30          I
         -RSM2763X
MIO       ASB1E    SX1    X1+100B     LEN = LEN+100B                                 RSM2763X     119     A
MIO                                                                                 RSM2763X     120     A
MIO                ALLOC  BLK,X1      ALLOC( BLK , LEN ) ; OL = OLD(L.BLK)           FCC0415       31     A
MIO                PRINT  ASB,(* INCREASE IN BLK = *,Z8),X1                          FCC0415       32     A
MIO                SB5    X2-1                                                       FCC0415       33     A
MIO                MX7    0                                                          FCC0415       34     A
MIO                SA7    B5+X3       [O.BLK+NL-1] = 0                               FCC0415       35     A
MIO                SB3    X1                                                         FCC0415       36     A
MIO                SB2    B5+B6       FWA = O.BLK-1+OL                               FCC0415       37     A
MIO                CALL   ABB         CALL ABB(FWA,LEN)  */ADD BLOCK TO LIST         FCC0415       38     A
MIO                SA1    ASBB                                                       FCC0415       39          I
         -RSM2763X
MIO                BX6    X1                                                         FCC0415       40          I
         -RSM2763X
MIO                SA6    =XTO        [TO] = DMBH    */RESTORE OVERFLOW EXIT         FCC0415       41          I

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

-RSM2763X

```
MIO                                                                                  FCC0415      42    A
MIO       ASB1D    SA2    ASBA                                                       FCC0415      43    A
MIO                SB3    X2+                                                        MIO        1117    A
MIO                                                                                  MIO        1118    A
MIO       ASB1A    SA1    LAS         BTW = LAS                                      MIO        1119    A
MIO                PRINT  ASB,(* --LAS = *Z20),LAS                                   MIO        1120    A
MIO                LX1    -BT.LKFP    I = LKF[BTW]                                   MIO        1121    A
MIO                SB5    X1                                                         MIO        1122    A
MIO                ZR     B5,ASB2A                                                   MIO        1123    A
MIO                                                                                  MIO        1124    A
MIO                SA2    X1+         BTW = [I]                                      MIO        1125    A
MIO                PRNTM  ASB,BTW,X2                                                 MIO        1126    A
MIO                                                                                  MIO        1127    A
MIO      *         SEARCH LIST FOR AVAILABLE SPACE                                   MIO        1128    A
MIO                                                                                  MIO        1129    A
MIO       ASB2     LX2    -BT.LENP                                                   MIO        1130    A
MIO                SB5    X2          L = LEN[BTW]                                   MIO        1131    A
MIO                LX2    BT.LENP-BT.LKFP                                            MIO        1132    A
MIO                GE     B5,B3,ASB3  IF L \ NL                                      MIO        1133    A
MIO                                                                                  MIO        1134    A
MIO                SB6    X2          I = LKF[BTW]                                   MIO        1135    A
MIO                SA2    X2          BTW = [I]                                      MIO        1136    A
MIO                PRNTM  ASB,BTW,X2                                                 MIO        1137    A
MIO                NZ     B6,ASB2     IF I " 0                                       MIO        1138    A
MIO                                                                                  MIO        1139    A
MIO      *         CALL GARBAGE COLLECTOR FOR BLK                                    MIO        1140    A
MIO                                                                                  MIO        1141    A
MIO       ASB2A    SX6    B3+                                                        MIO        1142    A
MIO                SA6    ASBA                                                       MIO        1143    A
MIO                CALL   CGB         CALL CGB      */GARBAGE COLLECTOR               MIO        1144    A
MIO                SA2    ASBA                                                       MIO        1145    A
MIO                SB3    X2+                                                        MIO        1146    A
MIO                EQ     ASB1                                                       MIO        1147    A
MIO                                                                                  MIO        1148    A
MIO      *         REMOVE BLOCK FROM AVAILABLE LIST                                  MIO        1149    A
MIO                                                                                  MIO        1150    A
MIO       ASB3     LX2    BT.LKFP                                                    MIO        1151    A
MIO                SB6    B5-B1                                                      MIO        1152    A
MIO                SB2    A2-B6       FBLK = I-L+1                                   MIO        1153    A
MIO                SA1    A2                                                         MIO        1154    A
MIO                CALL   RBB         CALL RBB(BTW)                                  MIO        1155    A
MIO                EQ     B5,B3,ASB5  IF L = NL                                      MIO        1156    A
MIO                                                                                  MIO        1157    A
MIO      *         ADD REMAINDER OF BLOCK TO AVAILABLE LIST                          MIO        1158    A
MIO                                                                                  MIO        1159    A
MIO       ASB4     SX4    B2+B1                                                      MIO        1160    A
MIO                MX7    0                                                          MIO        1161    A
MIO                SB2    B2+B3       FBLK = FBLK + NL                               MIO        1162    A
MIO                SB3    B5-B3       LX = L - NL                                    MIO        1163    A
MIO                SA7    B2-B1                                                      MIO        1164    A
MIO                CALL   ABB         CALL ABB(FWAX,LX)                              MIO        1165    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO                BX3    X4                                                  MIO        1166     A
MIO                EQ     ASB                                                 MIO        1167     A
MIO                                                                            MIO        1168     A
MIO       *        SET FWA OF ALLOCATED BLOCK                                  MIO        1169     A
MIO                                                                            MIO        1170     A
MIO        ASB5    SX3    B2+B1                                               MIO        1171     A
MIO                EQ     ASB                                                 MIO        1172     A
MIO                                                                            MIO        1173     A
MIO        ASB6    SA1    =7LTBL-OVF                                          RSM2763X    121     A
MIO                CALL   PUNT                                                RSM2763X    122     A
MIO                                                                            RSM2763X    123     A
MIO                                                                            RSM2763X    124     A
MIO        ASBA    BSS    1                                                   MIO        1174     A
MIO        ASBB    BSS    1                                                   FCC0415      44     A
MIO        ABB     TITLE  ABB - ADD BLOCK TO BLK LIST                         MIO        1175     A
MIO       **       ABB - ADD BLOCK TO BLK LIST                                 MIO        1176     A
MIO       *                                                                    MIO        1177     A
MIO       *        ENTRY  (B2) = FWA, FWA OF BLOCK (ADDRESS OF BHW)            MIO        1178     A
MIO       *               (B3) = LEN, LENGTH OF BLOCK (INCLUDING BHW AND BTW)  MIO        1179     A
MIO       *                                                                    MIO        1180     A
MIO       *        EXIT   BLOCK ADDED TO LINKED LIST                           MIO        1181     A
MIO       *                                                                    MIO        1182     A
MIO       *        CALLS  RBB.                                                 MIO        1183     A
MIO                                                                            MIO        1184     A
MIO        ABB     ROUTINE                                                     MIO        1185     A
MIO                                                                            MIO        1186          I
          -FCC0415
MIO                SA1    MX.AVS                                              MIO        1187     A
MIO                SB7    B3-B1                                               MIO        1188     A
MIO                SB4    B2+B7       ABTW = FBLK+L-1                         MIO        1189     A
MIO                SX7    X1+B3       MX.AVS = MX.AVS+LEN                     MIO        1190     A
MIO                SA7    A1                                                  MIO        1191     A
MIO                                                                            MIO        1192     A
MIO       *        GET BLOCK TRAILER WORDS FOR PRECEDING AND FOLLOWING BLOCKS. MIO        1193     A
MIO                                                                            MIO        1194     A
MIO                SA1    B2-B1       PBTW = [FBLK]                           MIO        1195     A
MIO                PRNTM  ABB,BTW,X1                                          MIO        1196     A
MIO                ZR     X1,ABB1     IF PBTW = 0   */NO PRECEDING BLOCK      MIO        1197     A
MIO                                                                            MIO        1198     A
MIO       *        COMBINE ADJACENT AVAILABLE BLOCKS                           MIO        1199     A
MIO                                                                            MIO        1200     A
MIO                LX1    59-BT.AVP                                           MIO        1201     A
MIO                PL     X1,ABB1     IF ^AV[PBTW]  */NOT AVAILABLE SPACE     MIO        1202     A
MIO                                                                            MIO        1203     A
MIO                LX1    1+BT.AVP-BT.LENP                                    MIO        1204     A
MIO                SB7    X1          PL = LEN[PBTW]                          MIO        1205     A
MIO                SA2    MX.AVS                                              MIO        1206     A
MIO                SB2    B2-B7       FBLK = FBLK-PL                          MIO        1207     A
MIO                SB3    B3+B7       L = L + PL                              MIO        1208     A
MIO                SX7    X2+B7                                               MIO        1209     A
MIO                LX1    BT.LENP                                             MIO        1210     A
MIO                SA7    A2                                                  MIO        1211     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO                CALL   RBB          CALL RBB(PBTW)                          MIO        1212    A
MIO                                                                             MIO        1213    A
MIO        ABB1    SA2    B4+B1        FBHW = [ABTW+1]                         MIO        1214    A
MIO                ZR     X2,ABB2      IF FBHW = 0   */NO FOLLOWING BLOCK      MIO        1215    A
MIO                                                                             MIO        1216    A
MIO                LX2    -BH.LENP     LS = LEN[FBHW]                          MIO        1217    A
MIO                SA3    X2+B4        FBTW = [ABTW+LS]                        MIO        1218    A
MIO                PRNTM  ABB,BTW,X3                                           MIO        1219    A
MIO                LX3    59-BT.AVP                                           MIO        1220    A
MIO                PL     X3,ABB2      IF ^AV[FBTW]  */NOT AVAILABLE SPACE     MIO        1221    A
MIO                                                                             MIO        1222    A
MIO                LX3    1+BT.AVP-BT.LENP                                    MIO        1223    A
MIO                SA2    MX.AVS                                              MIO        1224    A
MIO                SB7    X3           FL = LEN[FBTW]                          MIO        1225    A
MIO                SB3    B3+B7        L = L+FL                                MIO        1226    A
MIO                SB4    B4+B7        ABTW = ABTW+FL                          MIO        1227    A
MIO                SX7    X2+B7                                                MIO        1228    A
MIO                SA1    A3                                                   MIO        1229    A
MIO                SA7    A2                                                   MIO        1230    A
MIO                CALL   RBB          CALL RBB(FBTW)                          MIO        1231    A
MIO                                                                             MIO        1232    A
MIO        *       STORE NEW BLOCK HEADER AND TRAILER WORDS.                   MIO        1233    A
MIO                                                                             MIO        1234    A
MIO        ABB2    SX6    B3                                                   MIO        1235    A
MIO                MX1    1                                                    MIO        1236    A
MIO                LX6    BH.LENP      LEN[BHW] = LEN                          MIO        1237    A
MIO                LX1    1+BH.AVP     AV[BHW] = 1                             MIO        1238    A
MIO                BX6    X1+X6                                                MIO        1239    A
MIO                SB7    MIN.AB                                              MIO        1240    A
MIO                BX7    X6           BTW = BHW                               MIO        1241    A
MIO                SA6    B2           [FBLK] = BHW                            MIO        1242    A
MIO                PRNTM  ABB,BHW,X6                                           MIO        1243    A
MIO                LT     B7,B3,ABB3   IF MIN.AB < LEN                         MIO        1244    A
MIO                                                                             MIO        1245    A
MIO                SA7    B4           [ABTW] = BTW                            MIO        1246    A
MIO                PRNTM  ABB,BTW,X7                                           MIO        1247    A
MIO                EQ     ABB                                                  MIO        1248    A
MIO                                                                             MIO        1249    A
MIO        ABB3    SA3    LAS          OPTR = LAS                              MIO        1250    A
MIO                SX6    B4                                                   MIO        1251    A
MIO                LX6    BT.LKFP      LKF[NPTR] = ABTW                        MIO        1252    A
MIO                BX7    X3+X7        LKF[BTW] = LKF[OPTR]                    MIO        1253    A
MIO                SA6    A3           LAS = NPTR                              MIO        1254    A
MIO                PRINT  ABB,(* LAS = *Z20),LAS                               MIO        1255    A
MIO                LX3    -BT.LKFP                                            MIO        1256    A
MIO                SX3    X3           LINKF = LKF[OPTR]                       MIO        1257    A
MIO                ZR     X3,ABB4      IF LINKF = 0                            MIO        1258    A
MIO                                                                             MIO        1259    A
MIO                SA1    X3                                                   MIO        1260    A
MIO                MX0    -BT.LKBL                                            MIO        1261    A
MIO                LX0    BT.LKBP                                             MIO        1262    A
MIO                BX2    X0*X1        NWORD = LWORDF                          MIO        1263    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO                LX6    BT.LKBP-BT.LKFP   LKB[NWORDF] = ABTW                    MIO        1264     A
MIO                BX6    X2+X6                                                  MIO        1265     A
MIO                SA6    A1          [LINKF] = NWORD                            MIO        1266     A
MIO                PRNTM  ABB,BTW,X6                                             MIO        1267     A
MIO       ABB4     SA7    B4          [ABTW] = BTW                               MIO        1268     A
MIO                PRNTM  ABB,BTW,X7                                             MIO        1269     A
MIO                                                                             MIO        1270     A
MIO                EQ     ABB                                                    MIO        1271     A
MIO       RBB      TITLE  RBB - REMOVE BLOCK FROM BLK LIST                       MIO        1272     A
MIO      **        RBB - REMOVE BLOCK FROM BLK LIST                              MIO        1273     A
MIO      *                                                                      MIO        1274     A
MIO      *         ENTRY  (X1) = BTW, BLOCK TRAILER WORD                         MIO        1275     A
MIO      *                (A1) = ABTW, ADDRESS OF BLOCK TRAILER WORD             MIO        1276     A
MIO      *                                                                      MIO        1277     A
MIO      *         EXIT   BLOCK REMOVED FROM LINKED LIST                         MIO        1278     A
MIO      *                                                                      MIO        1279     A
MIO      *         CALLS  NONE.                                                  MIO        1280     A
MIO                                                                             MIO        1281     A
MIO       RBB      ROUTINE                                                       MIO        1282     A
MIO                SA2    MX.AVS                                                 MIO        1283     A
MIO                SB7    MIN.AB                                                 MIO        1284     A
MIO                LX1    -BT.LENP                                               MIO        1285     A
MIO                SX6    X1          L = LEN[BTW]                               MIO        1286     A
MIO                SB6    X1                                                     MIO        1287     A
MIO                IX7    X2-X6       MX.AVS = MX.AVS - L                        MIO        1288     A
MIO                SA7    A2                                                     MIO        1289     A
MIO                LT     B6,B7,RBB   IF L .LT. MIN.AB                           MIO        1290     A
MIO                                                                             MIO        1291     A
MIO                LX1    BT.LENP-BT.LKFP                                        MIO        1292     A
MIO                SB6    X1          LINKF = LKF[BTW]                           MIO        1293     A
MIO                LX1    BT.LKFP-BT.LKBP                                        MIO        1294     A
MIO                SB7    X1          LINKB = LKB[BTW]                           MIO        1295     A
MIO                NZ     B7,RBB1     IF LINKB " 0  */NOT FIRST ENTRY ON LIST    MIO        1296     A
MIO                                                                             MIO        1297     A
MIO                SA2    LAS                                                    MIO        1298     A
MIO                LX2    -BT.LKFP                                               MIO        1299     A
MIO                SX0    A1                                                     MIO        1300     A
MIO                IX0    X0-X2                                                  MIO        1301     A
MIO                NZ     X0,RBB      IF ABTW .NE. LKF[LAS]                      MIO        1302     A
MIO                                                                             MIO        1303     A
MIO                LX2    BT.LKFP                                                MIO        1304     A
MIO                EQ     RBB2                                                   MIO        1305     A
MIO                                                                             MIO        1306     A
MIO       RBB1     SA2    B7          LWORDB = [LINKB]                           MIO        1307     A
MIO                PRNTM  RBB,BTW,X2                                             MIO        1308     A
MIO       RBB2     MX1    -BT.LKFL                                               MIO        1309     A
MIO                SX0    B6                                                     MIO        1310     A
MIO                LX1    BT.LKFP                                                MIO        1311     A
MIO                BX2    X1*X2                                                  MIO        1312     A
MIO                LX0    BT.LKFP     LKF[LWORDB] = LINKF                        MIO        1313     A
MIO                BX6    X0+X2                                                  MIO        1314     A
MIO                SA6    A2          [LINKB] = LWORDB                           MIO        1315     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO                PRNTM  RBB,BTW,X6                                          MIO        1316     A
MIO                ZR     B6,RBB      IF LINKF = 0  */LAST ENTRY ON LIST      MIO        1317     A
MIO                                                                           MIO        1318     A
MIO                SA2    B6          LWORDF = [LINKF]                        MIO        1319     A
MIO                PRNTM  RBB,BTW,X2                                          MIO        1320     A
MIO                LX1    -BT.LKFP+BT.LKBP                                    MIO        1321     A
MIO                SX0    B7                                                  MIO        1322     A
MIO                BX2    X1*X2                                               MIO        1323     A
MIO                LX0    BT.LKBP     LKB[LWORDF] = LINKB                     MIO        1324     A
MIO                BX7    X0+X2                                               MIO        1325     A
MIO                SA7    A2          [LINKF] = LWORDF                        MIO        1326     A
MIO                PRNTM  RBB,BTW,X7                                          MIO        1327     A
MIO                EQ     RBB                                                 MIO        1328     A
MIO       CGB      TITLE  CGB - COLLECT GARBAGE IN BLK                        MIO        1329     A
MIO      **        CGB - COLLECT GARBAGE IN BLK                               MIO        1330     A
MIO      *                                                                   MIO        1331     A
MIO      *         ENTRY  NONE                                                MIO        1332     A
MIO      *                                                                   MIO        1333     A
MIO      *         EXIT   ALLOCATED BLOCKS MOVED TO A CONTIGUOUS AREA         MIO        1334     A
MIO      *                LIST OF AVAILABLE SPACE UPDATED                    MIO        1335     A
MIO      *                                                                   MIO        1336     A
MIO      *         CALLS  MVE=.                                               MIO        1337     A
MIO                                                                           MIO        1338     A
MIO       CGB      ENTRY. **                                                  MIO        1339          I
         -CCGNC
MIO       CGB      ENTRY. **,#                                                CCGNC       509     A
MIO                                                                           MIO        1340     A
MIO                PRINT  CGB,(* --MX.AVS = *Z8),MX.AVS                       MIO        1341     A
MIO                SA3    O.SEQ                                               MIO        1342     A
MIO                SA1    O.BIT                                               MIO        1343     A
MIO                SA2    O.BST                                               MIO        1344     A
MIO                SA4    O.BLK                                               MIO        1345     A
MIO                SB4    X3                                                  MIO        1346     A
MIO                SB2    X1+B1                                               MIO        1347     A
MIO                SB3    X2                                                  MIO        1348     A
MIO                SX3    X4+B1       TO = O.BLK+1                            MIO        1349     A
MIO                SA4    X4+B1       FROM = O.BLK+1; BHW = [FROM]            MIO        1350     A
MIO                PRNTM  CGB,BHW,X4                                          MIO        1351     A
MIO                SB6    60-BH.LENP                                          MIO        1352     A
MIO                                                                           MIO        1353     A
MIO      *         LOOP TO MOVE BLOCKS TO CONTIGUOUS AREA                     MIO        1354     A
MIO                                                                           MIO        1355     A
MIO       CGB1     ZR     X4,CGB5     IF BHW = 0    */ END OF TABLE          MIO        1356     A
MIO                                                                           MIO        1357     A
MIO                LX6    B6,X4       TBHW = BHW                              MIO        1358     A
MIO                SX2    A4          TFROM = FROM                            MIO        1359     A
MIO                SB5    X6          L = LEN[TBHW]                           MIO        1360     A
MIO                LX6    BH.LENP+59-BH.AVP                                   MIO        1361     A
MIO                SA4    A4+B5       FROM = FROM+L; BHW = [FROM]             MIO        1362     A
MIO                PRNTM  CGB,BHW,X4                                          MIO        1363     A
MIO                MI     X6,CGB1     IF AV[TBHW] = 1  */BLOCK AVAILABLE      MIO        1364     A
MIO                                                                           MIO        1365     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO                IX1    X3-X2                                                  MIO        1366     A
MIO                NZ     X1,CGB2     IF TFROM " TO */BLOCK MUST BE MOVED        MIO        1367     A
MIO                                                                              MIO        1368     A
MIO                SX3    X3+B5       TO = TO + L                                MIO        1369     A
MIO                EQ     CGB1                                                   MIO        1370     A
MIO                                                                              MIO        1371     A
MIO      *         UPDATE *BST*,*BIT* ENTRIES                                    MIO        1372     A
MIO                                                                              MIO        1373     A
MIO       CGB2     LX6    1+BH.AVP-BH.BIP                                        MIO        1374     A
MIO                MX1    -RI.FWAL                                               MIO        1375     A
MIO                SA4    X6+B2       RIW = BIT(BI[TBHW]+1)                      MIO        1376     A
MIO                PRNTM  CGB,RIW,X4                                             MIO        1377     A
MIO                LX1    RI.FWAP                                                MIO        1378     A
MIO                BX4    X1*X4                                                  MIO        1379     A
MIO                LX6    BH.BIP-BH.BSTP                                         MIO        1380     A
MIO                SX7    X3+B1       NFWA = TO + 1                              MIO        1381     A
MIO                SB7    X6          B = BST[TBHW]                              MIO        1382     A
MIO                LX7    RI.FWAP                                                MIO        1383     A
MIO                BX6    X4+X7       FWA[RIW] = NFWA                            MIO        1384     A
MIO                SA6    A4          BIT(BI[TBHW]+1) = RIW                      MIO        1385     A
MIO                PRNTM  CGB,RIW,X6                                             MIO        1386     A
MIO                MI     B7,CGB4     IF B < 0      */NO BST ENTRY               MIO        1387     A
MIO                                                                              MIO        1388     A
MIO                SA4    B3+B7       BAW = BST(B)                               MIO        1389     A
MIO                PRNTM  CGB,BAW,X4                                             MIO        1390     A
MIO                LX7    -RI.FWAP                                               FCC1498X      5     A
MIO                SB6    X2+B1       OFWA = TFROM + 1                           MIO        1391     A
MIO                NE     B4,B6,CGB3  IF O.SEQ " OFWA                            MIO        1392     A
MIO                                                                              MIO        1393     A
MIO                SA7    O.SEQ       O.SEQ = NFWA                               MIO        1394     A
MIO                                                                              MIO        1395     A
MIO       CGB3     LX1    BA.FWAP-RI.FWAP                                        MIO        1396     A
MIO                BX4    X1*X4                                                  MIO        1397     A
MIO                LX7    BA.FWAP-RI.FWAP                                        MIO        1398          I
          -FCC1498X
MIO                LX7    BA.FWAP                                                FCC1498X      6     A
MIO                BX6    X4+X7       FWA[BAW] = NFWA                            MIO        1399     A
MIO                PRNTM  CGB,BAW,X6                                             MIO        1400     A
MIO                SA6    A4          BST(B) = BAW                               MIO        1401     A
MIO                                                                              MIO        1402     A
MIO      *         UPDATE BLOCK TRAILER WORD AND MOVE BLOCK.                     MIO        1403     A
MIO                                                                              MIO        1404     A
MIO       CGB4     SX7    B5                                                     MIO        1405     A
MIO                LX7    BT.LENP     LEN[BTW] = L                               MIO        1406     A
MIO                SB7    B5-1                                                   MIO        1407     A
MIO                SA7    X2+B7       [TFROM+L-1] = BTW                          MIO        1408     A
MIO                PRNTM  CGB,BTW,X7                                             MIO        1409     A
MIO                SX1    B5                                                     MIO        1410     A
MIO                SB6    B5+X2       SFROM = TFROM + L                  002540  FCCA462       5     A
MIO                SB5    B5+X3       STO = STO + L                      002550  FCCA462       6     A
MIO                MOVE   X1,X2,X3    MOVE(L,TFROM,TO)                           MIO        1411     A
MIO                SA4    A4-B1       FROM = FROM+L; BHW = [FROM]                MIO        1412          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

          -FCCA462
MIO                SX3    A7+B1       TO = TO+L                                   MIO        1413          I
          -FCCA462
MIO                SA4    B6          FROM = SFROM                       002570   FCCA462       7     A
MIO                SX3    B5          TO = STO                           002580   FCCA462       8     A
MIO                SB6    60-BH.LENP                                              MIO        1414     A
MIO                EQ     CGB1                                                    MIO        1415     A
MIO                                                                               MIO        1416     A
MIO       *        UPDATE LIST OF AVAILABLE SPACE.                                MIO        1417     A
MIO                                                                               MIO        1418     A
MIO        CGB5    SA2    O.BLK                                                   MIO        1419     A
MIO                SA4    L.BLK                                                   MIO        1420     A
MIO                SB4    X4-1                                                    MIO        1421     A
MIO                SX4    X2+B4                                                   MIO        1422     A
MIO                IX7    X4-X3       L = FROM-TO                                 MIO        1423     A
MIO                MX6    0                                                       MIO        1424     A
MIO                SA6    X4          BLK(L.BLK) = 0                              MIO        1425     A
MIO                SA6    MX.AVS      MX.AVS = L    */MAX AVAILABLE SPACE IN BLK  MIO        1426     A
MIO                SB2    X3                                                      MIO        1427     A
MIO                SB3    X7                                                      MIO        1428     A
MIO                PRINT  CGB,(* MAX SPACE AFTER CGB = *Z8),B3                    MIO        1429     A
MIO                SA6    LAS         LAS = 0                                     MIO        1430     A
MIO                ZR     X7,CGB      IF L = 0  */ NO FREE SPACE                  FCCA509X      7     A
MIO                CALL   ABB         CALL ABB(TO,L)                              MIO        1431     A
MIO                EQ     CGB                                                     MIO        1432     A
MIO                                                                               MIO        1433     A
MIO        #ECS    IFNE   CT.ECS,0                                                MIO        1434     A
MIO        ASL     TITLE  ASL - ALLOCATE SPACE IN LCM                             MIO        1435     A
MIO       **       ASL - ALLOCATE SPACE IN LCM                                    MIO        1436     A
MIO       *                                                                       MIO        1437     A
MIO       *        ENTRY  (B2) = FBLK, FWA OF LCM BLOCK                           MIO        1438     A
MIO       *                (B5) = NL, NEW LENGTH REQUESTED                        MIO        1439     A
MIO       *                                                                       MIO        1440     A
MIO       *        EXIT   (X2) = FWA, FWA OF BLOCK ALLOCATED                      MIO        1441     A
MIO       *                (X7) = L, LENGTH OF BLOCK ALLOCATED                    MIO        1442     A
MIO       *                                                                       MIO        1443     A
MIO       *        CALLS  ABL,CGL,RBL.                                            MIO        1444     A
MIO                                                                               MIO        1445     A
MIO        ASL     ROUTINE                                                        MIO        1446     A
MIO                                                                               MIO        1447          I
          -FCC0415
MIO                ZR     B2,ASL1     IF FBLK = 0   */NO SPACE ALLOCATED          MIO        1448     A
MIO                                                                               MIO        1449     A
MIO       *        CHECK IF BLOCK CAN BE REWRITTEN IN SAME SPACE                  MIO        1450     A
MIO                                                                               MIO        1451     A
MIO                SX0    B2                                                      MIO        1452     A
MIO                RX1    X0,ASLA     BHW = [FLCM]                                MIO        1453     A
MIO                PRNTM  ASL,BHW,X1                                              MIO        1454     A
MIO                LX1    -BH.LENP                                                MIO        1455     A
MIO                SB3    X1          OL = LEN[BHW]                               MIO        1456     A
MIO                LE     B5,B3,ASL7  IF NL .LE. OL */REUSE BLOCK                 MIO        1457     A
MIO                                                                               MIO        1458     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO                CALL   ABL         CALL ABL(FBLK,OL)                          MIO        1459    A
MIO                                                                              MIO        1460    A
MIO      *         CHECK IF SPACE AVAILABLE AT O.LCM                             MIO        1461    A
MIO                                                                              MIO        1462    A
MIO       ASL1     SA2    O.LCM       FBLK = O.LCM                               MIO        1463    A
MIO                SA1    LCM.FL                                                 MIO        1464    A
MIO                SA3    MAX.LCM                                                MIO        1465    A
MIO                SB5    B5+LCM.XL   NL = NL + LCM.XL  */EXTRA LENGTH           MIO        1466    A
MIO                PRINT  ASL,(* --O.LCM,LCM.FL,MAX.LCM,LEN OF BLOCK = *4Z8),(X2, MIO        1467    A
MIO      ,X1,X3,B5)                                                              MIO        1468    A
MIO                SX6    X2+B5       L = O.LCM + NL                             MIO        1469    A
MIO                SX4    X1+B5       FL = LCM.FL + NL                           MIO        1470    A
MIO                IX7    X6-X1                                                  MIO        1471    A
MIO                MI     X7,ASL2     IF L .LT. LCM.FL                           MIO        1472    A
MIO                                                                              MIO        1473    A
MIO                SX7    X4+10007B   NFL = FL + 10007B                          MIO        1474    A
MIO                IX4    X7-X3                                                  MIO        1475    A
MIO                PL     X4,ASL3     IF NFL . GE. MAX.LCM                       MIO        1476    A
MIO                                                                              MIO        1477    A
MIO      *         REQUEST MORE STORAGE                                          MIO        1478    A
MIO                                                                              MIO        1479    A
MIO                AX7    9                                                      MIO        1480    A
MIO                LX7    39                                                     MIO        1481    A
MIO                SA7    ASLA+1                                                 MIO        1482    A
MIO                BX3    X6                                                     CCGA012       6    A
MIO                BX4    X2                                                     CCGA012       7    A
MIO                MEMORY LCM,ASLA+1,R                                           MIO        1483    A
MIO                BX6    X3                                                     CCGA012       8    A
MIO                BX2    X4                                                     CCGA012       9    A
MIO                SA3    ASLA+1                                                 MIO        1484    A
MIO                BX7    X3                                                     MIO        1485    A
MIO                AX7    30                                                     MIO        1486    A
MIO                SA7    LCM.FL                                                 MIO        1487    A
MIO                PRINT  ASL,(* LCM REQUEST - LCM.FL = *Z8),X7                   MIO        1488    A
MIO                                                                              MIO        1489    A
MIO       ASL2     SA6    A2          O.LCM = L                                  MIO        1490    A
MIO                SX7    B5                                                     MIO        1491    A
MIO                EQ     ASL                                                    MIO        1492    A
MIO                                                                              MIO        1493    A
MIO       ASL3     SA2    MX.AVL                                                 MIO        1494    A
MIO                AX1    2           L = LCM.FL / 4                             MIO        1495    A
MIO                SB4    X2                                                     MIO        1496    A
MIO                GT     B5,B4,ASL8  IF NL > MX.AVL                             MIO        1497    A
MIO                                                                              MIO        1498    A
MIO                IX6    X1-X2                                                  MIO        1499    A
MIO                MI     X6,ASL5     IF MX.AVL > L */GARBAGE COLLECT            MIO        1500    A
MIO                                                                              MIO        1501    A
MIO      *         SEARCH AVAILABLE LIST                                         MIO        1502    A
MIO                                                                              MIO        1503    A
MIO                SA1    LAL         BTW = LAL                                  MIO        1504    A
MIO                PRINT  ASL,(* --LAL = *Z20),LAL                               MIO        1505    A
MIO                LX1    -BT.LKFP    I = LKF[BTW]                               MIO        1506    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO                SX0    X1                                                  MIO        1507    A
MIO                ZR     X0,ASL8     IF LAL = 0                              MIO        1508    A
MIO                                                                            MIO        1509    A
MIO                RX2    X0,ASLA     BTW = [I]                               MIO        1510    A
MIO                PRNTM  ASL,BTW,X2                                          MIO        1511    A
MIO                                                                            MIO        1512    A
MIO       ASL4     LX2    -BT.LENP                                            MIO        1513    A
MIO                SB3    X2          L = LEN[BTW]                            MIO        1514    A
MIO                LX2    BT.LENP-BT.LKFP                                     MIO        1515    A
MIO                LE     B5,B3,ASL6  IF NL @ L                               MIO        1516    A
MIO                                                                            MIO        1517    A
MIO                SB6    X2          I = LKF[BTW]                            MIO        1518    A
MIO                SX0    X2                                                  MIO        1519    A
MIO                RX2    X0,ASLA     BTW = [I]                               MIO        1520    A
MIO                PRNTM  ASL,BTW,X2                                          MIO        1521    A
MIO                NZ     B6,ASL4                                             MIO        1522    A
MIO                EQ     ASL8                                                MIO        1523    A
MIO                                                                            MIO        1524         I
          -FCC0415
MIO                                                                            MIO        1525    A
MIO       ASL5     SX6    B5                                                  MIO        1526    A
MIO                SA6    ASLA                                                MIO        1527    A
MIO                CALL   CGL         CALL GCL                                MIO        1528    A
MIO                SA1    ASLA                                                MIO        1529    A
MIO                SB5    X1                                                  MIO        1530         I
          -FCC0391
MIO                SB5    X1-LCM.XL   NL = NL - LCM.XL                        FCC0391      10    A
MIO                EQ     ASL1                                                MIO        1531    A
MIO                                                                            MIO        1532    A
MIO      *         REMOVE BLOCK FROM AVAILABLE LIST                           MIO        1533    A
MIO                                                                            MIO        1534    A
MIO       ASL6     LX2    BT.LKFP                                             MIO        1535    A
MIO                BX1    X2                                                  MIO        1536    A
MIO                SB4    B1-B3                                               MIO        1537    A
MIO                SB2    X0+B4                                               MIO        1538    A
MIO                CALL   RBL         CALL RBL(BTW)                           MIO        1539    A
MIO                SX2    B2                                                  MIO        1540    A
MIO                SX7    B3                                                  MIO        1541    A
MIO                EQ     ASL                                                 MIO        1542    A
MIO                                                                            MIO        1543    A
MIO       ASL7     SX2    B2          FWA = FBLK                              MIO        1544    A
MIO                SX7    B3          L = OL                                  MIO        1545    A
MIO                EQ     ASL                                                 MIO        1546    A
MIO                                                                            MIO        1547    A
MIO       ASL8     SX2    -1                                                  MIO        1548    A
MIO                EQ     ASL                                                 MIO        1549    A
MIO                                                                            MIO        1550    A
MIO       ASLA     BSS    2                                                   MIO        1551    A
MIO       ABL      TITLE  ABL - ADD BLOCK TO LCM LIST                         MIO        1552    A
MIO      **        ABL - ADD BLOCK TO LCM LIST                                MIO        1553    A
MIO      *                                                                    MIO        1554    A
MIO      *         ENTRY  (B2) = FBLK, FWA OF BLOCK (ADDRESS OF BHW)          MIO        1555    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO      *                (B3) = L, LENGTH OF BLOCK (INCLUDING BHW AND BTW)
MIO      *
MIO      *         EXIT   BLOCK ADDED TO LINKED LIST
MIO      *
MIO      *         CALLS  RBL.
MIO
MIO       ABL      ROUTINE
MIO
         -FCC0415
MIO                SA1    MX.AVL
MIO                SB7    B3-B1
MIO                SB4    B2+B7       ABTW = FBLK+L-1
MIO                SX6    X1+B3       MX.AVL = MX.AVL + LEN
MIO                SA6    A1+
MIO
MIO      *         GET BLOCK TRAILER WORDS FOR PRECEDING AND FOLLOWING BLOCKS.
MIO
MIO                SX0    B2-B1
MIO                MI     X0,ABL1     IF APBTW < 0  */NO PRECEEDING BLOCK
MIO
MIO                RX1    X0,ABLA     PBTW = [FBLK-1]
MIO                PRNTM  ABL,BTW,X1
MIO
MIO      *         COMBINE ADJACENT AVAILABLE BLOCKS
MIO
MIO                LX1    59-BT.AVP
MIO                PL     X1,ABL1     IF ^AV[PBTW]  */NOT AVAILABLE SPACE
MIO
MIO                LX1    1+BT.AVP-BT.LENP
MIO                SB7    X1          PL = LEN[PBTW]
MIO                SA2    MX.AVL
MIO                SB2    B2-B7       FBLK = FBLK-PL
MIO                SB3    B3+B7       L = L + PL
MIO                SX7    X2+B7
MIO                LX1    BT.LENP
MIO                SA7    A2
MIO                CALL   RBL         CALL RBL(PBTW)
MIO
MIO       ABL1     SA1    O.LCM
MIO                SX0    B4+B1
MIO                IX1    X1-X0
MIO                ZR     X1,ABL2     IF AFBHW = 0  */NO FOLLOWING BLOCK
MIO
MIO                RX2    X0,ABLB     FBHW = [ABTW+1]
MIO                LX2    -BH.LENP    LS = LEN[FBHW]
MIO                SX0    X2+B4
MIO                RX3    X0,ABLB     FBTW = [ABTW+LS]
MIO                PRNTM  ABL,BTW,X3
MIO                LX3    59-BT.AVP
MIO                PL     X3,ABL2     IF ^AV[FBTW]  */NOT AVAILABLE SPACE
MIO
MIO                LX3    1+BT.AVP-BT.LENP
```

```
         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO                SB7    X3          FL = LEN[FBTW]                          MIO        1607     A
MIO                SA2    MX.AVL                                              MIO        1608     A
MIO                SB3    B3+B7       L = L+FL                                MIO        1609     A
MIO                SB4    B4+B7       ABTW = ABTW+FL                          MIO        1610     A
MIO                BX1    X3                                                  MIO        1611     A
MIO                SX7    X2+B7                                               MIO        1612     A
MIO                LX1    BT.LENP                                             MIO        1613     A
MIO                SA7    A2                                                  MIO        1614     A
MIO                CALL   RBL         CALL RBL(FBTW)                          MIO        1615     A
MIO                                                                           MIO        1616     A
MIO      *         STORE NEW BLOCK HEADER AND TRAILER WORDS.                  MIO        1617     A
MIO                                                                           MIO        1618     A
MIO       ABL2     SX6    B3                                                  MIO        1619     A
MIO                MX1    1                                                   MIO        1620     A
MIO                LX6    BH.LENP     LEN[BHW] = LEN                          MIO        1621     A
MIO                LX1    1+BH.AVP    AV[BHW] = 1                             MIO        1622     A
MIO                BX6    X1+X6                                               MIO        1623     A
MIO                SB7    MIN.AL                                              MIO        1624     A
MIO                BX7    X6          BTW = BHW                               MIO        1625     A
MIO                SX0    B2                                                  MIO        1626     A
MIO                WX6    X0,ABLA     [FBLK] = BHW                            MIO        1627     A
MIO                PRNTM  ABL,BHW,X6                                          MIO        1628     A
MIO                LT     B7,B3,ABL3  IF MIN.AL < LEN                        MIO        1629     A
MIO                                                                           MIO        1630     A
MIO                SX0    B4                                                  MIO        1631     A
MIO                WX7    X0,ABLA     [ABTW] = BTW                            MIO        1632     A
MIO                PRNTM  ABL,BTW,X7                                          MIO        1633     A
MIO                EQ     ABL                                                 MIO        1634     A
MIO                                                                           MIO        1635     A
MIO       ABL3     SA3    LAL         OPTR = LAL                              MIO        1636     A
MIO                SX6    B4                                                  MIO        1637     A
MIO                LX6    BT.LKFP     LKF[NPTR] = ABTW                        MIO        1638     A
MIO                BX7    X3+X7       LKF[BTW] = LKF[OPTR]                    MIO        1639     A
MIO                SA6    A3          LAS = NPTR                              MIO        1640     A
MIO                PRINT  ABL,(* LAL = *Z20),LAL                              MIO        1641     A
MIO                LX3    -BT.LKFP                                            MIO        1642     A
MIO                SX3    X3          LINKF = LKF[OPTR]                       MIO        1643     A
MIO                ZR     X3,ABL4     IF LINK = 0                             MIO        1644     A
MIO                                                                           MIO        1645     A
MIO                SX0    X3                                                  MIO        1646     A
MIO                RX1    X0,ABLA     LWORDF = [LINKF]                        MIO        1647     A
MIO                MX3    -BT.LKBL                                            MIO        1648     A
MIO                LX3    BT.LKBP                                             MIO        1649     A
MIO                BX2    X3*X1       NWORD = LWORDF                          MIO        1650     A
MIO                LX6    BT.LKBP-BT.LKFP   LKB[NWORD] = ABTW                MIO        1651     A
MIO                BX6    X2+X6                                               MIO        1652     A
MIO                WX6    X0,ABLA     [LINKF] = NWORD                         MIO        1653     A
MIO                PRNTM  ABL,BTW,X6                                          MIO        1654     A
MIO                                                                           MIO        1655     A
MIO       ABL4     SX0    B4                                                  MIO        1656     A
MIO                WX7    X0,ABLA     [ABTW] = BTW                            MIO        1657     A
MIO                PRNTM  ABL,BTW,X7                                          MIO        1658     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO                EQ     ABL                                                  MIO        1659     A
MIO                                                                             MIO        1660     A
MIO       ABLA     BSS    1                                                    MIO        1661     A
MIO       ABLB     BSS    1                                                    MIO        1662     A
MIO       RBL      TITLE  RBL - REMOVE BLOCK FROM LCM LIST                     MIO        1663     A
MIO      **        RBL - REMOVE  BLOCK FROM LCM LIST                           MIO        1664     A
MIO      *                                                                     MIO        1665     A
MIO      *         ENTRY  (X1) = BTW, BLOCK TRAILER WORD                       MIO        1666     A
MIO      *                                                                     MIO        1667     A
MIO      *         EXIT   BLOCK REMOVED FROM LINKED LIST                       MIO        1668     A
MIO      *                                                                     MIO        1669          I
         -FCC0415
MIO      *         CALLS  NONE.                                                MIO        1670          I
         -FCC0415
MIO                                                                             MIO        1671     A
MIO       RBL      ROUTINE                                                     MIO        1672     A
MIO                                                                             MIO        1673          I
         -FCC0415
MIO                SA2    MX.AVL                                               MIO        1674     A
MIO                SB7    MIN.AL                                               MIO        1675     A
MIO                LX1    -BT.LENP                                             MIO        1676     A
MIO                SX6    X1          L = LEN[BTW]                             MIO        1677     A
MIO                SB6    X1                                                   MIO        1678     A
MIO                IX7    X2-X6       MX.AVL = MX.AVL - L                      MIO        1679     A
MIO                LX1    BT.LENP-BT.LKFP                                      MIO        1680     A
MIO                SA7    A2                                                   MIO        1681     A
MIO                LT     B6,B7,RBL   IF L .LT. MIN.AB                         MIO        1682     A
MIO                                                                             MIO        1683     A
MIO                SB6    X1          LINKF = LKF[BTW]                         MIO        1684     A
MIO                LX1    BT.LKFP-BT.LKBP                                      MIO        1685     A
MIO                SB7    X1          LINKB = LKB[BTW]                         MIO        1686     A
MIO                NZ     B7,RBL1     IF LINKB " 0  */NOT FIRST ENTRY ON LIST  MIO        1687     A
MIO                                                                             MIO        1688     A
MIO                SA2    LAL                                                  MIO        1689     A
MIO                MX4    0                                                    MIO        1690     A
MIO                EQ     RBL2                                                 MIO        1691     A
MIO                                                                             MIO        1692     A
MIO       RBL1     SX0    B7                                                   MIO        1693     A
MIO                MX4    1                                                    MIO        1694     A
MIO                RX2    X0,RBLA     LWORDB = [LINKB]                         MIO        1695     A
MIO                PRNTM  RBL,BTW,X2                                           MIO        1696     A
MIO       RBL2     MX1    -BT.LKFL                                             MIO        1697     A
MIO                SX7    B6                                                   MIO        1698     A
MIO                LX1    BT.LKFP                                              MIO        1699     A
MIO                BX2    X1*X2                                                MIO        1700     A
MIO                LX7    BT.LKFP     LKF[LWORDB] = LINKF                      MIO        1701     A
MIO                BX6    X2+X7                                                MIO        1702     A
MIO                PL     X4,RBL3                                              MIO        1703     A
MIO                                                                             MIO        1704     A
MIO                WX6    X0,RBLA     [LINKB] = LWORDB                         MIO        1705     A
MIO                PRNTM  RBL,BTW,X6                                           MIO        1706     A
MIO                EQ     RBL4                                                 MIO        1707     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO                                                                    MIO        1708    A
MIO      RBL3     SA6    LAL                                           MIO        1709    A
MIO               PRINT  RBL,(* LAL = *Z20),LAL                        MIO        1710    A
MIO                                                                    MIO        1711    A
MIO      RBL4     BSS    0                                             MIO        1712        I
        -FCC0415
MIO               ZR     B6,RBL      IF LINKF = 0  */LAST ENTRY ON LIST MIO        1713        I
        -FCC0415
MIO                                                                    MIO        1714        I
        -FCC0415
MIO      RBL4     ZR     B6,RBL      IF LINKF = 0  */ LAST ENTRY ON LIST FCC0415     45    A
MIO               LX1    -BT.LKFP+BT.LKBP                              MIO        1715    A
MIO               SX0    B6                                            MIO        1716    A
MIO               RX2    X0,RBLA     LWORDF = [LINKF]                  MIO        1717    A
MIO               PRNTM  RBL,BTW,X2                                    MIO        1718    A
MIO               SX6    B7                                            MIO        1719    A
MIO               BX2    X1*X2                                         MIO        1720    A
MIO               LX6    BT.LKBP     LKB[LWORDF] = LINKB               MIO        1721    A
MIO               BX7    X6+X2                                         MIO        1722    A
MIO               WX7    X0,RBLA     [LINKF] = LWORDF                  MIO        1723    A
MIO               PRNTM  RBL,BTW,X7                                    MIO        1724    A
MIO               EQ     RBL                                           MIO        1725    A
MIO                                                                    MIO        1726    A
MIO      RBLA     BSS    1                                             MIO        1727    A
MIO      CGL      TITLE  CGL - COLLECT GARBAGE IN LCM                  MIO        1728    A
MIO      **       CGL - COLLECT GARBAGE IN LCM                         MIO        1729    A
MIO      *                                                             MIO        1730    A
MIO      *        ENTRY  NONE                                          MIO        1731    A
MIO      *                                                             MIO        1732    A
MIO      *        EXIT   ALLOCATED BLOCKS MOVED TO A CONTIGUOUS AREA   MIO        1733    A
MIO      *               LIST OF AVAILABLE SPACE UPDATED               MIO        1734    A
MIO      *                                                             MIO        1735        I
        -FCC0415
MIO      *                                                             MIO        1736    A
MIO      *        CALLS  MVL.                                          MIO        1737    A
MIO                                                                    MIO        1738    A
MIO      CGL      ROUTINE                                              MIO        1739    A
MIO                                                                    MIO        1740        I
        -FCC0415
MIO               PRINT  CGL,(* --MX.AVL " *Z8),MX.AVL                 MIO        1741        I
        -FCC0415
MIO               PRINT  CGL,(* -- MX.AVL =*Z8),MX.AVL                 FCC0415     46    A
MIO               SA3    O.LCM                                         MIO        1742    A
MIO               SA1    O.BIT                                         MIO        1743    A
MIO               MX6    0                                             MIO        1744    A
MIO               BX0    X3                                            MIO        1745    A
MIO               WX6    X0,CGLA     LCM(O.LCM) = 0 */LOOP TERMINATOR  MIO        1746    A
MIO               SB2    X1+B1                                         MIO        1747    A
MIO               SB6    60-BH.LENP                                    MIO        1748    A
MIO               MX0    0           FROM = 0                          MIO        1749    A
MIO               SX3    B0          TO = 0                            MIO        1750    A
MIO               RX4    X0,CGLA     BHW = LCM(FROM)                   MIO        1751    A

        0         1         2         3         4         5         6         7         8
        12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO                PRNTM  CGL,BHW,X4                                          MIO        1752     A
MIO                                                                           MIO        1753     A
MIO      *         LOOP TO MOVE BLOCKS TO CONTIGUOUS AREA.                     MIO        1754     A
MIO                                                                           MIO        1755     A
MIO       CGL1     ZR     X4,CGL5     IF BHW = 0    */END OF LCM              MIO        1756     A
MIO                                                                           MIO        1757     A
MIO                LX6    B6,X4       TBHW = BHW                              MIO        1758     A
MIO                BX2    X0          TFROM = FROM                            MIO        1759     A
MIO                SX1    X6          L = LEN[TBHW]                           MIO        1760     A
MIO                LX6    BH.LENP+59-BH.AVP                                   MIO        1761     A
MIO                IX0    X0+X1       FROM = FROM + L                         MIO        1762     A
MIO                RX4    X0,CGLA     BHW = LCM(FROM)                         MIO        1763     A
MIO                PRNTM  CGL,BHW,X4                                          MIO        1764     A
MIO                MI     X6,CGL1     IF AV[TBHW]     */BLOCK AVAILABLE       MIO        1765     A
MIO                                                                           MIO        1766     A
MIO                SB5    X1-2        LEN = L - 2                             MIO        1767     A
MIO                IX7    X3-X2                                               MIO        1768     A
MIO                NZ     X7,CGL2     IF TFROM " TO  */BLOCK MUST BE MOVED    MIO        1769     A
MIO                                                                           MIO        1770     A
MIO                IX3    X1+X3       TO = TO + L                             MIO        1771     A
MIO                EQ     CGL1                                                MIO        1772     A
MIO                                                                           MIO        1773     A
MIO      *         UPDATE *BIT* ENTRY AND MOVE BLOCK HEADER WORD.              MIO        1774     A
MIO                                                                           MIO        1775     A
MIO       CGL2     BX7    X4                                                  MIO        1776     A
MIO                LX6    1+BH.AVP-BH.BIP                                     MIO        1777     A
MIO                MX1    -RI.LCML                                            MIO        1778     A
MIO                SA4    X6+B2       RIW = BIT(BI[TBHW]+1)                   MIO        1779     A
MIO                PRNTM  CGL,RIW,X4                                          MIO        1780     A
MIO                SA7    CGLA                                                MIO        1781     A
MIO                LX1    RI.LCMP                                             MIO        1782     A
MIO                SX7    X3          NLCM = TO                               MIO        1783     A
MIO                BX4    X1*X4                                               MIO        1784     A
MIO                LX7    RI.LCMP                                             MIO        1785     A
MIO                BX7    X4+X7       RA[RIW] = NLCM                          MIO        1786     A
MIO                SA7    A4          BIT(BI[TBHW+1]) = RIW                   MIO        1787     A
MIO                PRNTM  CGL,RIW,X7                                          MIO        1788     A
MIO                BX0    X3                                                  MIO        1789     A
MIO                LX6    BH.BIP                                              MIO        1790     A
MIO                WX6    X0,CGLB     LCM(TO) = BHW                           MIO        1791     A
MIO                PRNTM  CGL,BHW,X6                                          MIO        1792     A
MIO                                                                           MIO        1793     A
MIO      *         MOVE BLOCK                                                  MIO        1794     A
MIO                                                                           MIO        1795     A
MIO                SX1    B5                                                  MIO        1796     A
MIO                SX2    X2+B1       TFROM = TFROM+1                         MIO        1797     A
MIO                SX3    X3+B1       TO = TO+1                               MIO        1798     A
MIO                CALL   MVL         MOVE(LEN,TFROM,TO)                      MIO        1799     A
MIO                                                                           MIO        1800     A
MIO      *         MOVE BLOCK TRAILER WORD.                                    MIO        1801     A
MIO                                                                           MIO        1802     A
MIO                SB4    B1+B1                                               MIO        1803     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO                SX7    X1+B4                                                MIO        1804     A
MIO                BX0    X3          TFROM = TFROM+L; TO = TO+L               MIO        1805     A
MIO                LX7    BT.LENP     LEN[BTW] = L                             MIO        1806     A
MIO                WX7    X0,CGLB     LCM(TO) = BTW                            MIO        1807     A
MIO                PRNTM  CGL,BTW,X7                                           MIO        1808     A
MIO                SA4    CGLA                                                 MIO        1809     A
MIO                SB6    60-BH.LENP                                           MIO        1810     A
MIO                SX0    X2+B1       FROM = TFROM+1                           MIO        1811     A
MIO                SX3    X3+B1       TO = TO+1                                MIO        1812     A
MIO                EQ     CGL1                                                 MIO        1813     A
MIO                                                                            MIO        1814     A
MIO      *         UPDATE O.LCM AND LIST OF AVAILABLE SPACE.                   MIO        1815     A
MIO                                                                            MIO        1816     A
MIO       CGL5     MX7    0                                                    MIO        1817     A
MIO                BX6    X3                                                   MIO        1818     A
MIO                SA7    LAL         LAL = 0                                  MIO        1819     A
MIO                SA6    O.LCM       O.LCM = TO                               MIO        1820     A
MIO                PRINT  CGL,(* --O.LCM = *Z8),O.LCM                          MIO        1821     A
MIO                SA7    MX.AVL      MX.AVL = 0                               MIO        1822     A
MIO                EQ     CGL                                                  MIO        1823     A
MIO                                                                            MIO        1824     A
MIO       CGLA     BSS    1                                                    MIO        1825     A
MIO       CGLB     BSS    1                                                    MIO        1826     A
MIO       MVL      TITLE  MVL - MOVE BLOCK OF LCM DATA                         MIO        1827     A
MIO      **        MVL - MOVE BLOCK OF LCM DATA                                MIO        1828     A
MIO      *                                                                     MIO        1829     A
MIO      *         ENTRY  (X1) = WORD COUNT                                    MIO        1830     A
MIO      *                (X2) = SOURCE FWA                                    MIO        1831     A
MIO      *                (X3) = DESTINATION FWA                               MIO        1832     A
MIO                                                                            MIO        1833     A
MIO       MVL      ROUTINE                                                     MIO        1834     A
MIO                SB7    X1                                                   MIO        1835     A
MIO                IX6    X2-X3                                                MIO        1836     A
MIO                SB6    100B        MOVE 100B WORDS AT A TIME.               MIO        1837     A
MIO                SA0    =XB=FBV                                              MIO        1838     A
MIO                LE     B7,B6,MVL2  IF ONLY ONE MOVE NEEDED.                 MIO        1839     A
MIO                                                                            MIO        1840     A
MIO                SB5    B6                                                   MIO        1841     A
MIO                PL     X6,MVL1     IF MOVE DOWN.                            MIO        1842     A
MIO                                                                            MIO        1843     A
MIO                SB5    B7-B6                                                MIO        1844     A
MIO                SX2    X2+B5       PREPARE FOR UPWARD MOVE.                 MIO        1845     A
MIO                SX3    X3+B5                                                MIO        1846     A
MIO                SB5    -B6                                                  MIO        1847     A
MIO                                                                            MIO        1848     A
MIO       MVL1     SX0    X2          READ BLOCK.                              MIO        1849     A
MIO                                                                            MIO        1850          I
          -F7540FT
MIO       #MD      IFLT   CT.CPU,75                                            MIO        1851          I
          -S3143FC -F7540FT
MIO       #MD      IFLT   CT.CPU,76                                            S3143FC      55          I
          -F7540FT

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO      +         RE     B6                                                  MIO        1852          I
         -F7540FT
MIO      -         EQ     ERRECSR                                             MIO        1853          I
         -F7540FT
MIO       #MD      ELSE                                                       MIO        1854          I
         -F7540FT
MIO                RL     B6                                                  MIO        1855          I
         -F7540FT
MIO       #MD      ENDIF                                                      MIO        1856          I
         -F7540FT
MIO                                                                           F7540FT      64     A
MIO                RL     B6                                                  F7540FT      65     A
MIO                                                                           MIO        1857     A
MIO                SX2    X2+B5                                               MIO        1858     A
MIO                SX0    X3          WRITE BLOCK.                            MIO        1859     A
MIO                                                                           MIO        1860          I
         -F7540FT
MIO       #MD      IFLT   CT.CPU,75                                           MIO        1861          I
         -S3143FC  -F7540FT
MIO       #MD      IFLT   CT.CPU,76                                           S3143FC      56          I
         -F7540FT
MIO      +         WE     B6                                                  MIO        1862          I
         -F7540FT
MIO      -         EQ     ERRECSW                                             MIO        1863          I
         -F7540FT
MIO       #MD      ELSE                                                       MIO        1864          I
         -F7540FT
MIO                WL     B6                                                  MIO        1865          I
         -F7540FT
MIO       #MD      ENDIF                                                      MIO        1866          I
         -F7540FT
MIO                                                                           F7540FT      66     A
MIO                WL     B6                                                  F7540FT      67     A
MIO                                                                           MIO        1867     A
MIO                SB7    B7-B6       COUNT WORDS.                            MIO        1868     A
MIO                SX3    X3+B5                                               MIO        1869     A
MIO                GT     B7,B6,MVL1  LOOP.                                   MIO        1870     A
MIO                                                                           MIO        1871     A
MIO       MVL2     LE     B7,B0,MVL   RETURN IF FINISHED.                     MIO        1872     A
MIO                                                                           MIO        1873     A
MIO                SB6    B7          SETUP FOR LAST MOVE.                    MIO        1874     A
MIO                SB5    B7                                                  MIO        1875     A
MIO                EQ     MVL1                                                MIO        1876     A
MIO                                                                           MIO        1877     A
MIO       #ECS     ENDIF                                                      MIO        1878     A
MIO                TTL    MIO - MASS STORAGE RANDOM I/O (BASIC I/O)           MIO        1879     A
MIO       SBD      TITLE  SBD - SAVE BLOCK TO DISK                            MIO        1880     A
MIO      **        SBD - SAVE BLOCK TO DISK                                   MIO        1881     A
MIO      *                                                                   MIO        1882     A
MIO      *         ENTRY  (X4) = FSCM, SCM FWA OF BLOCK                       MIO        1883     A
MIO      *                (X5) = LEN, LENGTH OF BLOCK                         MIO        1884     A
MIO      *                                                                   MIO        1885     A

         0        1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO      *         EXIT    (X1) = RIW, RANDOM INDEX WORD                          MIO        1886     A
MIO      *                 2/AT,1/0,18/LEN,9/0,30/RA.                             MIO        1887     A
MIO      *                                                                        MIO        1888     A
MIO      *         CALLS   WRITER, PUTW.                                          MIO        1889     A
MIO                                                                               MIO        1890     A
MIO       SBD      ENTRY. **                                                      MIO        1891          I
         -CCGNC
MIO       SBD      ENTRY. **,#                                                    CCGNC       510     A
MIO                                                                               MIO        1892     A
MIO      *         WRITE RECORD TO DISK.                                          MIO        1893     A
MIO                                                                               MIO        1894     A
MIO                SA2     O.DISK      DA = O.DISK   */DISK ADDRESS.               MIO        1895     A
MIO                BX6     X5                                                     MIO        1896     A
MIO                SA6     SBDB                                                   MIO        1897     A
MIO                                                                               MIO        1898     A
MIO       #RM      IFEQ    CP#RM,0                                                MIO        1899     A
MIO                                                                               MIO        1900     A
MIO      *         6000 MACE I/O                                                  MIO        1901     A
MIO                                                                               MIO        1902     A
MIO                SX3     X5                                                     MIO        1903     A
MIO                AX3     6                                                      MIO        1904     A
MIO                SX3     X3+B1       RL=LEN/100B+1 */RECORD LENGTH              MIO        1905     A
MIO                IX6     X2+X3                                                  MIO        1906     A
MIO                SA6     A2          O.DISK = DA + RL                           MIO        1907     A
MIO                                              */SET OPT FET FOR WRITE          MIO        1908     A
MIO                SETFET OPT,X4,X5,WRITE   SET OPT FET FOR WRITE                 MIO        1909     A
MIO                SX6     SBDA                                                   MIO        1910     A
MIO                SA6     =XF.OPT+6                                              MIO        1911     A
MIO                WRITER =XF.OPT,R   WRITE RECORD                                MIO        1912     A
MIO                SA2     SBDA                                                   MIO        1913     A
MIO       #RM      ELSE                                                           MIO        1914     A
MIO                                                                               MIO        1915     A
MIO      *         7000 RECORD MANAGER I/O                                        MIO        1916     A
MIO                                                                               MIO        1917     A
MIO                PUTW    =XFI.OPT,X4,X5,,X2        PUTW(FSCM,LEN,,DA)           MIO        1918     A
MIO                FETCH   =XFI.OPT,WA=X3                                         MIO        1919     A
MIO                BX6     X3                                                     MIO        1920     A
MIO                SA6     A2          O.DISK = WA                                MIO        1921     A
MIO       #RM      ENDIF                                                          MIO        1922     A
MIO                                                                               MIO        1923     A
MIO      *         FORM RANDOM INDEX WORD.                                        MIO        1924     A
MIO                                                                               MIO        1925     A
MIO                SA5     SBDB                                                   MIO        1926     A
MIO                LX5     RI.LENP     LEN[RIW] = LEN                             MIO        1927     A
MIO                LX2     RI.RAP      RA[RIW] = FSCM                             MIO        1928     A
MIO                BX1     X2+X5       RIW           */RANDOM INDEX WORD          MIO        1929     A
MIO                LX5     -RI.LENP                                               MIO        1930     A
MIO                EQ      SBD                                                    MIO        1931     A
MIO                                                                               MIO        1932     A
MIO       SBDA     BSS     1                                                      MIO        1933          I
         -FCC0415
MIO       SBDA     BSSZ    1                                                      FCC0415      47     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO        SBDB     BSS    1                                                    MIO        1934    A
MIO        GBD      TITLE  GBD - GET BLOCK FROM DISK                            MIO        1935    A
MIO       **        GBD - GET BLOCK FROM DISK                                   MIO        1936    A
MIO       *                                                                     MIO        1937    A
MIO       *         ENTRY  (X2) = FSCM, SCM FWA OF BLOCK                        MIO        1938    A
MIO       *                (X4) = DA, DISK ADDRESS                              MIO        1939    A
MIO       *                (X5) = LEN, LENGTH OF BLOCK                          MIO        1940    A
MIO       *                                                                     MIO        1941    A
MIO       *         EXIT   RECORD READ                                          MIO        1942    A
MIO       *                                                                     MIO        1943    A
MIO       *         CALLS  READ, GETW.                                          MIO        1944    A
MIO                                                                             MIO        1945    A
MIO        GBD      ENTRY. **                                                   MIO        1946          I
          -CCGNC
MIO        GBD      ENTRY. **,#                                                 CCGNC       511    A
MIO                                                                             MIO        1947    A
MIO       *         READ RECORD FROM DISK.                                      MIO        1948    A
MIO                                                                             MIO        1949    A
MIO        #RM      IFEQ   CP#RM,0                                              MIO        1950    A
MIO                                                                             MIO        1951    A
MIO       *         6000 MACE I/O                                               MIO        1952    A
MIO                                                                             MIO        1953    A
MIO                 MX6    -RI.RAL                                              MIO        1954    A
MIO                 BX7    -X6*X4                                               MIO        1955    A
MIO                 SA7    =XF.OPT+6   [F.OPT+6] = DA                           MIO        1956    A
MIO                 SETFET OPT,X2,X5,READ                                       MIO        1957    A
MIO                 READ   =XF.OPT,R   READ RECORD                              MIO        1958    A
MIO        #RM      ELSE                                                        MIO        1959    A
MIO                                                                             MIO        1960    A
MIO       *         7000 RECORD MANAGER I/O                                     MIO        1961    A
MIO                                                                             MIO        1962    A
MIO                 GETW   =XFI.OPT,X2,X5,,X4        GETW(FSCM,LEN,,DA)         MIO        1963    A
MIO        #RM      ENDIF                                                       MIO        1964    A
MIO                                                                             MIO        1965    A
MIO                 EQ     GBD                                                  MIO        1966    A
MIO                                                                             MIO        1967    A
MIO        #ECS     IFNE   CT.ECS,0                                             MIO        1968    A
MIO        SBL      TITLE  SBL - SAVE BLOCK TO LCM                              MIO        1969    A
MIO       **        SBL - SAVE BLOCK TO LCM                                     MIO        1970    A
MIO       *                                                                     MIO        1971    A
MIO       *         ENTRY  (X2) = FLCM, LCM FWA OF BLOCK                        MIO        1972    A
MIO       *                (X4) = FSCM, SCM FWA OF BLOCK                        MIO        1973    A
MIO       *                (X5) = LEN, LENGTH OF BLOCK                          MIO        1974    A
MIO       *                                                                     MIO        1975    A
MIO       *         EXIT   (X1) = RIW, RANDOM INDEX WORD                        MIO        1976    A
MIO       *                2/AT,1/0,18/LEN,9/0,30/RA.                           MIO        1977    A
MIO       *                                                                     MIO        1978          I
          -FCC0415
MIO       *         CALLS  NONE.                                                MIO        1979          I
          -FCC0415
MIO                                                                             MIO        1980    A
MIO        SBL      ROUTINE                                                     MIO        1981    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO                                                                                  MIO        1982     A
MIO      *         FORM RANDOM INDEX WORD.                                           MIO        1983     A
MIO                                                                                  MIO        1984     A
MIO                SA0    X4          FSCM = SCM FWA OF BLOCK                        MIO        1985     A
MIO                BX0    X2          FLCM = LCM FWA OF BLOCK                        MIO        1986     A
MIO                SB2    X5          L = LEN                                        MIO        1987     A
MIO                MX1    1                                                          MIO        1988     A
MIO                SB3    1777B                                                      MIO        1989          I
         -FCC0448
MIO                SB3    BLK.SIZ                                                    FCC0448      15     A
MIO                LX1    1+RI.ATP    AT[RIW] = 1   */AT = LCM                       MIO        1990     A
MIO                LX2    RI.RAP      RA[RIW] = FLCM                                 MIO        1991     A
MIO                BX1    X1+X2                                                      MIO        1992     A
MIO                LX5    RI.LENP     LEN[RIW] = LEN                                 MIO        1993     A
MIO                BX1    X1+X5       RIW           */RANDOM INDEX WORD              MIO        1994     A
MIO                LX5    -RI.LENP                                                   MIO        1995     A
MIO                                                                                  MIO        1996     A
MIO      *         WRITE BLOCK TO LCM.                                               MIO        1997     A
MIO                                                                                  MIO        1998     A
MIO                LE     B2,B3,SBL2  IF L. LE. 1777B                                MIO        1999          I
         -FCC0448
MIO                LE     B2,B3,SBL2  IF L .LE. BLK.SIZ                              FCC0448      16     A
MIO                                                                                  MIO        2000     A
MIO       SBL1     BSS    0                                                          MIO        2001          I
         -F7540FT
MIO                                                                                  MIO        2002          I
         -F7540FT
MIO       #MD      IFLT   CT.CPU,75                                                  MIO        2003          I
         -S3143FC  -F7540FT
MIO       #MD      IFLT   CT.CPU,76                                                  S3143FC      57          I
         -F7540FT
MIO      +         WE     B3          MOVE(1777B,FSCM,FLCM)                          MIO        2004          I
         -FCC0448  -F7540FT
MIO      +         WE     B3          MOVE(BLK.SIZ,FSCM,FLCM)                        FCC0448      17          I
         -F7540FT
MIO      -         EQ     ERRECSW                                                    MIO        2005          I
         -F7540FT
MIO       #MD      ELSE                                                              MIO        2006          I
         -F7540FT
MIO                WL     B3          MOVE(1777B,FSCM,FLCM)                          MIO        2007          I
         -FCC0448  -F7540FT
MIO                WL     B3          MOVE(BLK.SIZ,FSCM,FLCM)                        FCC0448      18          I
         -F7540FT
MIO       #MD      ENDIF                                                             MIO        2008          I
         -F7540FT
MIO       SBL1     WL     B3          MOVE(BLK.SIZ,FSCM,FLCM)                        F7540FT      68     A
MIO                                                                                  MIO        2009     A
MIO                SB2    B2-B3       L = L - 1777B                                  MIO        2010          I
         -FCC0448
MIO                SA0    A0+B3       FSCM = FSCM + 1777B                            MIO        2011          I
         -FCC0448
MIO                SX0    X0+B3       FLCM = FLCM + 1777B                            MIO        2012          I

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
          -FCC0448
MIO                GT     B2,B3,SBL1  IF L .GT. 1777B                          MIO         2013          I
          -FCC0448
MIO                SB2    B2-B3       L = L - BLK.SIZ                          FCC0448       19     A
MIO                SA0    A0+B3       FSCM = FSCM + BLK.SIZ                    FCC0448       20     A
MIO                SX0    X0+B3       FLCM = FLCM + BLK.SIZ                    FCC0448       21     A
MIO                GT     B2,B3,SBL1  IF L.GT. BLK.SIZ                         FCC0448       22     A
MIO                                                                            MIO         2014     A
MIO       SBL2     BSS    0                                                    MIO         2015          I
          -F7540FT
MIO                                                                            MIO         2016          I
          -F7540FT
MIO       #MD      IFLT   CT.CPU,75                                            MIO         2017          I
          -S3143FC -F7540FT
MIO       #MD      IFLT   CT.CPU,76                                            S3143FC       58          I
          -F7540FT
MIO       +        WE     B2          MOVE(L,FSCM,FLCM)                        MIO         2018          I
          -F7540FT
MIO       -        EQ     ERRECSW                                              MIO         2019          I
          -F7540FT
MIO       #MD      ELSE                                                        MIO         2020          I
          -F7540FT
MIO                WL     B2          MOVE(L,FSCM,FLCM)                        MIO         2021          I
          -F7540FT
MIO       #MD      ENDIF                                                       MIO         2022          I
          -F7540FT
MIO       SBL2     WL     B2          MOVE(L,FSCM,FLCM)                        F7540FT       69     A
MIO                                                                            MIO         2023     A
MIO                EQ     SBL                                                  MIO         2024     A
MIO                                                                            MIO         2025     A
MIO       ERRECSW  SX1    =C$ ECS WRITE ERROR$                                 MIO         2026     A
MIO                EQ     =XFTNABT                                             MIO         2027          I
          -CCG1
MIO                CALL   CMPLABT                                              CCG1          14          I
          -CCGNC
MIO                CALL   HE$ABT                                               CCGNC        512     A
MIO       GBL      TITLE  GBL - GET BLOCK FROM LCM                             MIO         2028     A
MIO       **       GBL - GET BLOCK FROM LCM                                    MIO         2029     A
MIO       *                                                                    MIO         2030     A
MIO       *        ENTRY  (X2) = FSCM, SCM FWA OF BLOCK                        MIO         2031     A
MIO       *               (X4) = FLCM, LCM FWA OF BLOCK                        MIO         2032     A
MIO       *               (X5) = LEN, LENGTH OF BLOCK                          MIO         2033     A
MIO       *                                                                    MIO         2034     A
MIO       *        EXIT   BLOCK TRANSFER COMPLETE                              MIO         2035     A
MIO       *                                                                    MIO         2036          I
          -FCC0415
MIO       *        CALLS  NONE.                                                MIO         2037          I
          -FCC0415
MIO                                                                            MIO         2038     A
MIO       GBL      ROUTINE                                                     MIO         2039     A
MIO                                                                            MIO         2040     A
MIO       *        READ BLOCK FROM LCM.                                        MIO         2041     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO                                                                                  MIO        2042     A
MIO                SA0    X2          FSCM = SCM FWA OF BLOCK                        MIO        2043     A
MIO                BX0    X4          FLCM = LCM FWA OF BLOCK                        MIO        2044     A
MIO                SB2    X5          L = LEN                                        MIO        2045     A
MIO                SB3    1777B                                                      MIO        2046          I
          -FCC0448
MIO                SB3    BLK.SIZ                                                    FCC0448      23     A
MIO                LT     B2,B3,GBL2  IF L .LE. 1777B                                MIO        2047          I
          -FCC0448
MIO                LT     B2,B3,GBL2  IF L .LE. BLK.SIZ                              FCC0448      24     A
MIO                                                                                  MIO        2048     A
MIO        GBL1    BSS    0                                                          MIO        2049          I
          -F7540FT
MIO                                                                                  MIO        2050          I
          -F7540FT
MIO        #MD     IFLT   CT.CPU,75                                                  MIO        2051          I
          -S3143FC -F7540FT
MIO        #MD     IFLT   CT.CPU,76                                                  S3143FC      59          I
          -F7540FT
MIO       +        RE     B3          MOVE(1777B,FLCM,FSCM)                          MIO        2052          I
          -FCC0448 -F7540FT
MIO       +        RE     B3          MOVE(BLK.SIZ,FLCM,FSCM)                        FCC0448      25          I
          -F7540FT
MIO       -        EQ     ERRECSR                                                    MIO        2053          I
          -F7540FT
MIO        #MD     ELSE                                                              MIO        2054          I
          -F7540FT
MIO                RL     B3          MOVE(1777B,FLCM,FSCM)                          MIO        2055          I
          -FCC0448 -F7540FT
MIO                RL     B3          MOVE(BLK.SIZ,FLCM,FSCM)                        FCC0448      26          I
          -F7540FT
MIO        #MD     ENDIF                                                             MIO        2056          I
          -F7540FT
MIO        GBL1    RL     B3          MOVE(BLK.SIZ,FLCM,FSCM)                        F7540FT      70     A
MIO                                                                                  MIO        2057     A
MIO                SB2    B2-B3       L = L - 1777B                                  MIO        2058          I
          -FCC0448
MIO                SA0    A0+B3       FSCM = FSCM + 1777B                            MIO        2059          I
          -FCC0448
MIO                SX0    X0+B3       FLCM = FLCM + 1777B                            MIO        2060          I
          -FCC0448
MIO                GE     B2,B3,GBL1  IF L . GT. 1777B                               MIO        2061          I
          -FCC0448
MIO                SB2    B2-B3       L = L - BLK.SIZ                                FCC0448      27     A
MIO                SA0    A0+B3       FSCM = FSCM + BLK.SIZ                          FCC0448      28     A
MIO                SX0    X0+B3       FLCM = FLCM + BLK.SIZ                          FCC0448      29     A
MIO                GE     B2,B3,GBL1  IF L .GT. BLK.SIZ                              FCC0448      30     A
MIO                                                                                  MIO        2062     A
MIO        GBL2    BSS    0                                                          MIO        2063          I
          -F7540FT
MIO                                                                                  MIO        2064          I
          -F7540FT

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

```
MIO       #MD      IFLT   CT.CPU,75                                          MIO        2065        I
         -S3143FC  -F7540FT
MIO       #MD      IFLT   CT.CPU,76                                          S3143FC      60        I
         -F7540FT
MIO      +         RE     B2          MOVE(L,FLCM,FSCM)                      MIO        2066        I
         -F7540FT
MIO      -         EQ     ERRECSR                                            MIO        2067        I
         -F7540FT
MIO       #MD      ELSE                                                      MIO        2068        I
         -F7540FT
MIO                RL     B2          MOVE(L,FLCM,FSCM)                      MIO        2069        I
         -F7540FT
MIO       #MD      ENDIF                                                     MIO        2070        I
         -F7540FT
MIO       GBL2     RL     B2          MOVE(L,FLCM,FSCM)                      F7540FT      71     A
MIO                                                                          MIO        2071     A
MIO                EQ     GBL                                                MIO        2072     A
MIO                                                                          MIO        2073     A
MIO       ERRECSR  SX1    =C$ ECS READ ERROR$                                MIO        2074     A
MIO                EQ     =XFTNABT                                           MIO        2075        I
         -CCG1
MIO                CALL   CMPLABT                                            CCG1         15        I
         -CCGNC
MIO                CALL   HE$ABT                                             CCGNC       513     A
MIO                                                                          MIO        2076     A
MIO       #ECS     ENDIF                                                     MIO        2077     A
MIO                                                                          MIO        2078     A
MIO                END                                                       MIO        2079     A
```

SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - MIO

IDENTIFIER	TOTAL	ACTIVE
MIO	2079	1862
FCC0391	4	4
FCC0415	43	31
FCC0448	18	12
FCC0534	5	3
FCC0634	8	7
S3143FC	7	0
FEAT155F	1	1
S3143FCA	1	0
FCC1123	4	4
FCC1352	2	1
F7540FT	35	30
FCC1498X	2	2
RSM2763X	59	59
CCG1	14	3
CCG7	13	6
CCGNC	33	33
CCG18	4	4
FCCA462	4	4

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MIO

                F2950CC                     5         5
                CCGA012                     5         5
                FCCA509X                    1         1
                F1037CC                    37        37

                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FBV

      FBV       *DECK     FBV                                                   FBV            1     A
      FBV                 IDENT  FBV                                            FBV            2     A
      FBV        FBV      TITLE  FBV - FORM BIT VECTORS ( USE/DEF AND LIVE EXIT ) FBV          3     A
      FBV       *CALL     SSTCALL                                               FBV            4     A
      FBV        B=FBV    RPVDEF                                                FBV            5     A
      FBV                 PASS2TM                                               FBV            6           I
                -CCG1
      FBV                                                                       FBV            7           I
                -CCG1
      FBV        FBV      SPACE  2                                              FBV            8     A
      FBV       *IF       DEF,FBV,1                                             FBV            9     A
      FBV                 TRACER CLI                                            FBV           10     A
      FBV       **        FBV - FORM BIT VECTORS ( PHASE 2 OF OPT=2 PASS 2 ).   FBV           11     A
      FBV       *         READ PROGRAM BLOCKS TO FORM *USE, DBU AND USE* VECTORS. FBV         12     A
      FBV       *         USE ULLMAN ALGORITHM TO FORM BLOCK *LIVE EXIT* BIT VECTORS. FBV     13     A
      FBV       *                                                               FBV           14     A
      FBV       *         NOTE - THE SPACE OCCUPIED BY *FBV* IS USED AS A BUFFER FOR FBV      15     A
      FBV       *         LCM TO LCM TRANSFERS BY *MIO*.                        FBV           16     A
      FBV                                                                       FBV           17     A
      FBV       **        DEBUGING SNAP LABELS                                  FBV           18     A
      FBV       *                                                               FBV           19     A
      FBV       *         TRACER (FBV,SNO,CLI)                                  FBV           20     A
      FBV        FBV      SPACE  2                                              FBV           21     A
      FBV                 TABLES BIT,BVT,CFT,RND,SEQ,TXT                        FBV           22           I
                -CCG1
      FBV                 ECHO   1,Z=(BIT,BVT,CFT,RND,SEQ,TXT)                  CCG1          16           I
                -CCGNC
      FBV                 EXT    O.Z,L.Z                                        CCG1          17           I
                -CCGNC
      FBV                                                                       FBV           23           I
                -CCGNC
      FBV                 EXT    BN,BVL,M.BBL,NPB,O.IST                         FBV           24           I
                -CCGNC
      FBV        FBV      SPACE  3                                              FBV           25           I
                -CCGNC
      FBV       **        FBV.ERR - REQUESTED BLOCK IS NOT SAME AS BLOCK RECEIVED FROM FBV    26           I
                -CCGNC
      FBV       *         I/O ROUTINES.  THIS IS INDICATIVE OF A BUG IN *MIO* OR THE FBV      27           I
                -CCGNC
      FBV       *         OPERATING SYSTEM ( *READNS* FUNCTION ).               FBV           28           I
                -CCGNC
      FBV                                                                       FBV           29           I
                -CCGNC
      FBV        FBV.ERR  SX1    =C$ FTN/FBV - BLOCK READ ERROR$                FBV           30           I
                -CCG1     -CCGNC

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FBV

```
FBV                EQ     =XFTNABT                                          FBV          31          I
          -CCG1    -CCGNC
FBV        FBV.ERR SX1    =C$  FBV - BLOCK READ ERROR$                      CCG1         18          I
          -CCGNC
FBV                CALL   CMPLABT                                           CCG1         19          I
          -CCGNC
FBV                TABLES (BIT,BVT,CFT,RND,SEQ,TXT)                         CCGNC       514     A
FBV                EXT#   (M.BBL,BN,NPB,O.IST,BVL)                          CCGNC       515     A
FBV                                                                         CCGNC       516     A
FBV        FBV     SPACE  2                                                 FBV          32     A
FBV       *CALL    CFTBLD                                                   FBV          33     A
FBV        FBV     EJECT                                                    FBV          34     A
FBV       **       FBV - FORM BIT VECTORS                                   FBV          35     A
FBV                                                                         FBV          36     A
FBV        FBV     ENTRY. **                                                FBV          37          I
          -CCGNC
FBV        FBV     ENTRY. **,#                                              CCGNC       517     A
FBV                SA1    M.BBL                                             FBV          38     A
FBV                ALLOC  TXT,X1+2    ALLOC( TXT , M.BBL + 2 )              FBV          39     A
FBV                SX7    X2+B1       O.SEQ = O.TXT + 1                     FBV          40     A
FBV                SA7    O.SEQ                                             FBV          41     A
FBV                SX6    B1+B1       BN = 2                                FBV          42     A
FBV                                                                         FBV          43     A
FBV       *        READ PROGRAM BLOCKS SEQUENTIALLY AND FORM BIT VECTORS    FBV          44     A
FBV                                                                         FBV          45     A
FBV        FBV1    SA1    O.BIT                                             FBV          46     A
FBV                SX6    X6+2        BN = BN + 2                           FBV          47     A
FBV                SB2    X6                                                FBV          48     A
FBV                SA2    X1+B2       BIW = BIT(BN)                         FBV          49     A
FBV                BX7    X2                                                FBV          50     A
FBV                SA5    A2+B1       RIW = BIT(BN+1)                       FBV          51     A
FBV                SA3    O.SEQ                                             FBV          52     A
FBV                ZR     X5,FBV3     IF RIW = 0    */ END OF PROGRAM BLOCKS FBV          53     A
FBV                SA6    BN                                                FBV          54     A
FBV                SA7    BIW                                               FBV          55     A
FBV                CALL   RNB         READ NEXT BLOCK                       FBV          56          I
          -CCGNC
FBV                CALL   RNB#        READ NEXT BLOCK                       CCGNC       518     A
FBV                SA2    BIW                                               FBV          57     A
FBV                SA5    O.SEQ                                             FBV          58     A
FBV                LX2    59-BI.RBP                                         FBV          59     A
FBV                PL     X2,FBV2     IF ^RB[BIW]   */ DEAD CODE            FBV          60     A
FBV                SA4    BN                                                FBV          61     A
FBV                SA1    X5                                                FBV          62     A
FBV                BX6    X4-X1                                             FBV          63     A
FBV                SB2    X6                                                FBV          64     A
FBV                NZ     B2,FBV.ERR  IF BN[SEQ] " BN  */ READNS ERROR      FBV          65          I
          -CCGNC
FBV                NZ     B2,*+1S17   IF BN[SEQ] " BN */ READNS ERROR       CCGNC       519     A
FBV                                                                         FBV          66     A
FBV                CALL   FUD         FORM USE/DEF BIT VECTORS              FBV          67          I
          -CCGNC

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FBV

```
FBV                CALL   FUD#        FORM USE/DEF BIT VECTORS                    CCGNC        520     A
FBV                                                                               FBV           68     A
FBV       FBV2     SA1    BN                                                      FBV           69     A
FBV                SX6    X1+                                                     FBV           70     A
FBV                EQ     FBV1                                                    FBV           71     A
FBV                                                                               FBV           72     A
FBV       FBV3     SX7    0                                                       FBV           73     A
FBV                SA7    L.TXT       L.TXT = 0                                   FBV           74     A
FBV                                                                               FBV           75     A
FBV                RJ     SNO         SET NODE ORDER                              FBV           76     A
FBV                                                                               FBV           77     A
FBV                RJ     CLI         COMPUTE LIVE EXIT BIT VECTORS               FBV           78     A
FBV                                                                               FBV           79     A
FBV                IFEQ   CP#RM,0,1                                               FBV           80          I
          -CCG1
FBV                RECALL =XF.OPT     WAIT UNTIL READ NS FINISHED                 FBV           81     A
FBV                SX6    0                                                       FBV           82     A
FBV                SA6    L.RND       L.RND = 0                                   FBV           83     A
FBV                SA6    L.CFT       L.CFT = 0                                   FBV           84     A
FBV                                                                               FBV           85     A
FBV                IF     DEF,/DEBUG/CLI,1                                        FBV           86     A
FBV                DCALL  PRNTLX,([O.BVT],[O.BIT],NPB)                            FBV           87     A
FBV                                                                               FBV           88     A
FBV                EQ     FBV                                                     FBV           89     A
FBV                                                                               FBV           90     A
FBV       BIW      BSS    1           SAVED COPY OF BLOCK INDEX WORD              FBV           91     A
FBV       SNO      TITLE  SNO - SET NODE ORDER BY A DEPTH FIRST SEARCH            FBV           92     A
FBV      **        SNO - SET NODE ORDER USING A DEPTH FIRST SEARCH OF THE FLOW    FBV           93     A
FBV      *         GRAPH TO ASSIGN THE *NUM* ORDER TO THE BLOCKS IN THE ORDER     FBV           94     A
FBV      *         IN WHICH THEY ARE LAST VISITED.                                FBV           95     A
FBV                                                                               FBV           96     A
FBV      *         TEMPORARY TABLE FOR DEPTH FIRST SEARCH / *NUM* COMPUTATION     FBV           97     A
FBV                                                                               FBV           98     A
FBV                DESCRIBE I.,60     TTW                                         FBV           99     A
FBV       TREE     DEFINE 1           IN TREE                                     FBV          100     A
FBV                DEFINE 11                                                      FBV          101     A
FBV       SUI      DEFINE 18          SUCCESSOR INDEX                             FBV          102     A
FBV       NS       DEFINE 12          N.SUCC NOT VISITED                          FBV          103     A
FBV       BN       DEFINE 18          BLOCK NUMBER OF NUM(I)                      FBV          104     A
FBV                                                                               FBV          105     A
FBV      *         REGISTER ASSIGNMENTS                                           FBV          106     A
FBV                                                                               FBV          107     A
FBV       C0       MICRO  1,,/B7/     O.CFT                                       FBV          108     A
FBV       EI       MICRO  1,,/B6/     EIB                                         FBV          109     A
FBV       NO       MICRO  1,,/B5/     NUM COUNTER                                 FBV          110     A
FBV       ST       MICRO  1,,/B4/     STK - PUSHDOWN STACK                        FBV          111     A
FBV       NS       MICRO  1,,/B3/     N.SUCC                                      FBV          112     A
FBV       N        MICRO  1,,/X7/     NODE NUMBER                                 FBV          113     A
FBV                                                                               FBV          114     A
FBV       SNO      ROUTINE                                                        FBV          115     A
FBV                SA5    NPB                                                     FBV          116     A
FBV                SX2    X5+B1                                                   FBV          117     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FBV

```
FBV              LX1    B1,X2                                                  FBV        118    A
FBV              ALLOC  CFT,X1      ALLOC( CFT , 2*(NPB+1) )                   FBV        119    A
FBV              S"C0"  X2          C0 = O.CFT                                 FBV        120    A
FBV              SA1    O.IST                                                  FBV        121    A
FBV              SB2    X5          N = NPB                                    FBV        122    A
FBV              S"EI"  X1+3        EIB = O.IST + 3                            FBV        123    A
FBV              SX6    2000B                                                  FBV        124    A
FBV              SA5    "EI"+B1     EI = EIB + 1                               FBV        125    A
FBV              LX6    I.NSP                                                  FBV        126    A
FBV              MX0    1                                                      FBV        127    A
FBV              LX0    1+I.TREEP                                              FBV        128    A
FBV              BX6    X0+X6                                                  FBV        129    A
FBV              SA6    "C0"        CI = C0;  [CI] = 0  */ EXIT NODE HAS NO SUC FBV        130    A
FBV                                                                            FBV        131    A
FBV       *      SETUP AUX TABLE WITH SUCCESSOR INFORMATION                     FBV        132    A
FBV                                                                            FBV        133    A
FBV       SNO1   LX5    -EI.SUIP                                               FBV        134    A
FBV              UX4    "NS",X5                                                FBV        135    A
FBV              SX3    X4          SI = SUI[EI]                               FBV        136    A
FBV              SA5    A5+B1       EI = EI + 1                                FBV        137    A
FBV              PX6    "NS",X3                                                FBV        138    A
FBV              LX6    I.SUIP                                                 FBV        139    A
FBV              SB2    B2-B1       N = N - 1                                  FBV        140    A
FBV              SA6    A6+B1       CI = CI + 1;  [CI] = TTW(0,SI,NS,0)        FBV        141    A
FBV              NZ     B2,SNO1     IF N " 0                                   FBV        142    A
FBV                                                                            FBV        143    A
FBV              S"ST"  A6+B1       ST = CI + 1   */ STACK ADDRESS             FBV        144    A
FBV              MX7    0                                                      FBV        145    A
FBV              SA7    "ST"        [ST] = 0                                   FBV        146    A
FBV              S"NO"  B1          NO = 1                                     FBV        147    A
FBV              S"N"   1           N = 1         */ START WITH PROG ENTRY BLOCK FBV        148    A
FBV                                                                            FBV        149    A
FBV       *      VISIT NODE *N*                                                FBV        150    A
FBV                                                                            FBV        151    A
FBV       SNO2   SA5    "C0"+"N"    TW = C(N)                                  FBV        152    A
FBV              BX6    X0+X5       TREE[TW] = 1  */ INDICATE IN TREE          FBV        153    A
FBV              SA6    A5                                                     FBV        154    A
FBV              LX6    -I.SUIP                                                FBV        155    A
FBV              UX5    "NS",X6                                                FBV        156    A
FBV              SB2    "EI"+X6                                                FBV        157    A
FBV                                                                            FBV        158    A
FBV       *      VISIT SUCCESSORS                                              FBV        159    A
FBV                                                                            FBV        160    A
FBV       SNO3   ZR     "NS",SNO4   IF NS[TW] = 0 */ IF ALL SUCC VISITED       FBV        161    A
FBV              S"NS"  "NS"-B1     NS[TW] = NS[TW] - 1                        FBV        162    A
FBV              SA3    B2+"NS"     ET = EIB + SUI[TW] + NS[TW]                FBV        163    A
FBV              LX3    -ET.SUCCP   I = SUCC[ET]                               FBV        164    A
FBV              SA2    "C0"+X3                                                FBV        165    A
FBV              LX2    59-I.TREEP                                             FBV        166    A
FBV              MI     X2,SNO3     IF TREE[C(I)] */ IF SUCC IN TREE           FBV        167    A
FBV                                                                            FBV        168    A
FBV              PX6    "NS",X6                                                FBV        169    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FBV

```
FBV                LX6    I.SUIP                                                  FBV        170    A
FBV                SA6    A6                                                      FBV        171    A
FBV                                                                               FBV        172    A
FBV                S"ST"  "ST"+B1     ST = ST + 1   */ STACK NODE N                FBV        173    A
FBV                R=     X7,"N"                                                  FBV        174    A
FBV                SA7    "ST"        [ST]  = N                                   FBV        175    A
FBV                S"N"   X3          N = I         */ SET TO VISIT SUCCESSOR     FBV        176    A
FBV                EQ     SNO2                                                    FBV        177    A
FBV                                                                               FBV        178    A
FBV      *         LAST VISIT TO NODE, ASSIGN NUM(NO) = BLOCK NUMBER               FBV        179    A
FBV                                                                               FBV        180    A
FBV       SNO4     PX6    "NS",X6                                                 FBV        181    A
FBV                LX6    I.SUIP                                                  FBV        182    A
FBV                SA6    A6                                                      FBV        183    A
FBV                                                                               FBV        184    A
FBV                SA1    "C0"+"NO"                                               FBV        185    A
FBV                R=     X7,"N"                                                  FBV        186    A
FBV                LX7    I.BNP                                                   FBV        187    A
FBV                BX6    X1+X7       BN[C(NO)] = N */ SET ORDER OF LAST VISIT    FBV        188    A
FBV                SA6    A1                                                      FBV        189    A
FBV                S"NO"  "NO"+B1     NO = NO + 1                                 FBV        190    A
FBV                SA4    "ST"                                                    FBV        191    A
FBV                S"N"   X4          N = [ST]      */ FETCH LAST FROM STACK      FBV        192    A
FBV                S"ST"  "ST"-B1     ST = ST - 1                                 FBV        193    A
FBV                NZ     "N",SNO2    IF N " 0                                    FBV        194    A
FBV                                                                               FBV        195    A
FBV                MX6    0                                                       FBV        196    A
FBV                SA6    A6+B1       C(NO) = 0     */ TABLE TERMINATOR           FBV        197    A
FBV                SX7    "NO"+B1                                                 FBV        198    A
FBV                TRACE  SNO,CFT                                                 FBV        199    A
FBV                SA7    L.CFT       L.CFT = NO + 1                              FBV        200    A
FBV                EQ     SNO                                                     FBV        201    A
FBV       CLI      TITLE  CLI - COMPUTE LIVE EXIT INFORMATION                     FBV        202    A
FBV      **        CLI - COMPUTE LIVE EXIT INFORMATION ( BIT VECTORS ).           FBV        203    A
FBV      *                COMPUTE THE LIVE EXIT ( LX ) BIT VECTORS FOR EACH      FBV        204    A
FBV      *         PROGRAM BLOCK.   THE METHOD IS TAKEN FROM THE ALGORITHMS       FBV        205    A
FBV      *         DESCRIBED IN THE PAPER "ANALYSIS OF A SIMPLE ALGORITHM FOR     FBV        206    A
FBV      *         GLOBAL FLOW PROBLEM" BY HECHT AND ULLMAN, PAGES 207 - 217 IN   FBV        207    A
FBV      *         THE ACM CONFRENCE RECORD, OCT 73 SIGACT/SIGPLAN MEETING.       FBV        208    A
FBV                                                                               FBV        209    A
FBV       BI       MICRO  1,,/B5/     O.BIT                                       FBV        210    A
FBV       EI       MICRO  1,,/B6/     EDGE INDEX TABLE BASE                       FBV        211    A
FBV       BV       MICRO  1,,/B7/     O.BVT                                       FBV        212    A
FBV                                                                               FBV        213    A
FBV       CLI      ROUTINE                                                        FBV        214    A
FBV                SA2    O.IST                                                   FBV        215    A
FBV                SA2    X2+2        GBM = [O.IST+2]                             FBV        216    A
FBV                LX2    -GB.ILIP                                                FBV        217    A
FBV                ALLOC  RND,X2+3    ALLOC( RND , ILI[GBW]+3 )                   FBV        218    A
FBV                SA4    O.CFT                                                   FBV        219    A
FBV                SA1    O.BVT                                                   FBV        220    A
FBV                SA7    X2          CS = [O.RND]  */ LOOP CONTROL STORE INDEX   FBV        221    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FBV

```
FBV                SA3    O.IST                                                    FBV        222    A
FBV                SA0    X2+B1       C0 = CS + 1                                  FBV        223    A
FBV                SA5    X4+B1       NI = [O.CFT] + 1;  I = BN[NI]                FBV        224    A
FBV                SA2    O.BIT                                                    FBV        225    A
FBV                MX0    1                                                        FBV        226    A
FBV                S"BI"  X2                                                       FBV        227    A
FBV                S"EI"  X3+3        EIB = O.IST + 3  */ EDGE INDEX BASE          FBV        228    A
FBV                S"BV"  X1                                                       FBV        229    A
FBV                                                                                FBV        230    A
FBV       *        SETUP THE LOOP CONTROL VECTOR FOR THE LIVE EXIT COMPUTATION     FBV        231    A
FBV                                                                                FBV        232    A
FBV        CLI1    SA4    "EI"+X5     EI = EIB + I                                 FBV        233    A
FBV                SB4    "BI"+X5                                                  FBV        234    A
FBV                LX4    -EI.SUIP                                                 FBV        235    A
FBV                UX6    B2,X4       NS = NS[EI]                                  FBV        236    A
FBV                SA3    B4+X5                                                    FBV        237    A
FBV                SA2    "EI"+X4     SI = EIB + SUI[EI]                           FBV        238    A
FBV                LX3    -BI.BVIP    VI = BVI[BIT(2*I)]                           FBV        239    A
FBV                SX7    "BV"+X3     BVA = O.BVT + VI                             FBV        240    A
FBV                SA7    A7+B1       CS = CS + 1;  [CS] = BVA                     FBV        241    A
FBV                                                                                FBV        242    A
FBV        CLI2    LX2    -ET.SUCCP   J = SUCC[SI]                                 FBV        243    A
FBV                SB3    "BI"+X2                                                  FBV        244    A
FBV                SA1    B3+X2                                                    FBV        245    A
FBV                SA2    A2+B1       SI = SI + 1                                  FBV        246    A
FBV                SB2    B2-B1       NS = NS - 1                                  FBV        247    A
FBV                LX1    -BI.BVIP                                                 FBV        248    A
FBV                SX6    "BV"+X1     BVA = O.BVT + BVI[BIT(2*J)]                  FBV        249    A
FBV                BX7    X0+X6                                                    FBV        250    A
FBV                SA7    A7+1        CS = CS + 1;  [CS] = 1S59 + BVA              FBV        251    A
FBV                NZ     B2,CLI2     IF NS " 0                                    FBV        252    A
FBV                                                                                FBV        253    A
FBV                SA5    A5+1        NI = NI + 1;  I = BN[NI]                     FBV        254    A
FBV                NZ     X5,CLI1     IF [NI] " 0                                  FBV        255    A
FBV                MX7    0                                                        FBV        256    A
FBV                SA7    A7+B1       CS = CS + 1;  [CS] = 0  */ TERMINATOR        FBV        257    A
FBV        CLI     SPACE  1,10                                                     FBV        258    A
FBV       *        NOW COMPUTE THE LIVE EXIT VECTORS FOR EACH PROGRAM BLOCK        FBV        259    A
FBV                                                                                FBV        260    A
FBV                SA4    BVL                                                      FBV        261    A
FBV                SB2    B0          WI = 0        */ WORD INDEX                  FBV        262    A
FBV                SB3    X4+B1       VL = BVL + 1  */ LIMIT                       FBV        263    A
FBV                SB4    X4+B1       WIF = VL      */ WI TO *UBD* VECTOR          FBV        264    A
FBV                SB5    B3+B3                                                    FBV        265    A
FBV                SB5    B5+B3       WIL = 3*VL    */ WI TO LX VECTOR             FBV        266    A
FBV                SB6    B0          N = 0         */ N.ITERATIONS                FBV        267    A
FBV                                                                                FBV        268    A
FBV        CLI3    SA5    A0+         CI = C0;  BVA = [CI]                         FBV        269    A
FBV                SX0    0           CHNG = 0                                     FBV        270    A
FBV                                                                                FBV        271    A
FBV        CLI4    SA4    X5+B5       LXA = BVA + WIL;  OLX = [LXA]                FBV        272    A
FBV                SA5    A5+1        CI = CI + 1;  BVA = [CI]  */ BV ADDR OF SUC  FBV        273    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN FBV

```
FBV                MX7    0           LX = 0                                                  FBV        274     A
FBV                                                                                            FBV        275     A
FBV      *         LX = ! ( ( LX & ^DEF ) ! UBD ) OVER SUCCESSORS OF THE NODE.                  FBV        276     A
FBV                                                                                            FBV        277     A
FBV       CLI5     SA1    X5+B5       LXS = [BVA+WIL]                                         FBV        278     A
FBV                SA2    X5+B2       DVS = [BVA+WI]                                          FBV        279     A
FBV                SA3    X5+B4       UFS = [BVA+WIF]                                         FBV        280     A
FBV                SA5    A5+B1       CI = CI + 1;  BVA = [CI]                                FBV        281     A
FBV                BX6    -X2*X1                                                              FBV        282     A
FBV                NO                                                                         FBV        283     A
FBV                BX1    X6+X3                                                               FBV        284     A
FBV                BX7    X1+X7       LX = LX ! ((LXS&^DVS)!UFS)                              FBV        285     A
FBV                MI     X5,CLI5     IF [CI] < 0   */ NEXT IS A SUCC ADDR                    FBV        286     A
FBV                                                                                            FBV        287     A
FBV                BX6    X4-X7                                                               FBV        288     A
FBV                SA7    A4          [LXA] = LX                                              FBV        289     A
FBV                BX0    X0+X6       CHNG = CHNG ! XOR(OLX,LX)                               FBV        290     A
FBV                NZ     X5,CLI4     IF [CI] " 0   */ NOT END OF CONTROL VECTOR              FBV        291     A
FBV                                                                                            FBV        292     A
FBV      *         TEST FOR CONVERGENCE OF LX VECTORS                                          FBV        293     A
FBV                                                                                            FBV        294     A
FBV                CX6    X0                                                                  FBV        295     A
FBV                SB6    B6+B1       N = N + 1                                               FBV        296     A
FBV                NZ     X6,CLI3     IF COUNT( CHNG ) " 0                                    FBV        297     A
FBV                                                                                            FBV        298     A
FBV      *         ADVANCE TO NEXT BIT VECTOR WORD                                             FBV        299     A
FBV                                                                                            FBV        300     A
FBV                SB2    B2+B1       WI = WI + 1                                             FBV        301     A
FBV                SB4    B4+B1       WIF = WIF + 1                                           FBV        302     A
FBV                SB5    B5+B1       WIL = WIL + 1                                           FBV        303     A
FBV                SB6    B0          N = 0                                                   FBV        304     A
FBV                LT     B2,B3,CLI3  IF WI < VL                                              FBV        305     A
FBV                EQ     CLI                                                                 FBV        306     A
FBV                                                                                            FBV        307     A
FBV                IFLT   *-FBV,100B,1                                                        FBV        308     A
FBV                BSS    FBV+100B-*                                                          FBV        309     A
FBV                                                                                            FBV        310     A
FBV                END                                                                        FBV        311     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - FBV

          IDENTIFIER                TOTAL    ACTIVE

          FBV                         311       294
          CCG1                          4         0
          CCGNC                         7         7

          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

GPO       *DECK     GPO                                                                        GPO          1     A
GPO                 IDENT  GPO                                                                GPO          2     A
GPO        GPO      TITLE  GPO - GLOBAL PROGRAM OPTIMIZATION                                  GPO          3     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO      *CALL     SSTCALL                                                GPO          4    A
GPO       B=GPO    RPVDEF                                                 GPO          5    A
GPO                PASS2TM                                                GPO          6         I
         -CCG1
GPO                                                                       GPO          7         I
         -CCG1
GPO      *IF       DEF,GPO,2                                              GPO          8    A
GPO                TRACER (GPO,FUD,RDD,AUV,MIP,FXI,SHB,MPB)               GPO          9    A
GPO                TRACER (CIF,FIM)                                       GPO         10    A
GPO       GPO      SPACE  2                                               GPO         11    A
GPO      **        GPO - GLOBAL PROGRAM OPTIMIZATION                      GPO         12    A
GPO      *                                                                GPO         13    A
GPO      *         S. I. JASIK - CDC - SUNNYVALE - WINTER 73              GPO         14    A
GPO      *                                                                GPO         15    A
GPO      *         ON ENTRY TO *GPO* THE FIRST PASS OF FIVE HAS BEEN PERFORMED  GPO    16    A
GPO      *         BY *PRE*.  IT CONSISTS OF SUBDIVIDING THE INTERMEDIATE *TEXT*  GPO  17         I
         -CCGNC
GPO      *         INTO BASIC BLOCKS AND WRITING THEM TO A RANDOM FILE,   GPO         18         I
         -CCGNC
GPO      *         BY THE BRIDGE.  IT CONSISTS OF SUBDIVIDING THE INTERMEDIATE  CCGNC 521    A
GPO      *         *TEXT* IN BASIC BLOCKS AND WRITING THEM TO A RANDOM FILE,  CCGNC 522    A
GPO      *         COLLECTING CONTROL FLOW INFORMATION, AND BUILDING A DICTIONARY  GPO  19    A
GPO      *         OF REFERENCED VARIABLES FOR USE/DEFINITION ANALYSIS.   GPO         20    A
GPO      *         IN THE INITIALIZATION OF *GPO* THE USE/DEF DICTIONARY IS  GPO      21    A
GPO      *         RESOLVED AND THE CONTROL FLOW INFORMATION PROCESSED TO FIND  GPO   22    A
GPO      *         LOOPS, ETC.                                            GPO         23    A
GPO      *           PASS 2 READS THE RANDOM FILE SEQUENTIALLY TO FORM THE  GPO       24    A
GPO      *         USE/DEF BIT VECTORS FOR EACH BLOCK, AND THEN PROCESS"S THEM  GPO   25    A
GPO      *         TO FORM THE *LIVE EXIT* VECTORS, WHICH ARE USED TO ELEMINATE  GPO  26    A
GPO      *         DEAD DEFINITIONS AND TO DETERMINE ENTRY/EXIT CONDITIONS  GPO       27    A
GPO      *         DURING REGISTER ASSIGNMENT.                            GPO         28    A
GPO      *           PASS 3  PROCESSES THE PROGRAM LOOPS IN AN INNER TO OUTER  GPO    29    A
GPO      *         ORDER, MOVING OUT INVARIANT AND STRENGTH REDUCABLE CODE,  GPO      30    A
GPO      *         ASSIGNING REGISTERS ACCROSS THE LOOP AND PROPAGATING THE  GPO      31    A
GPO      *         ENTRY/EXIT CONDITIONS OUT.  FINALLY THE REMAINING STRAIGHT  GPO   32    A
GPO      *         LINE CODE IS PROCESSED AS A SERIES OF EXTENDED BASIC BLOCKS  GPO  33    A
GPO      *         AND CODED.                                             GPO         34    A
GPO      *           PASS 4  REORDERS THE CODED BLOCKS TO BRING THEM BACK TO  GPO     35    A
GPO      *         SOURCE PROGRAM ORDER, ASSIGNS ADDRESS"S, AND PACKS THE  GPO       36    A
GPO      *         COMPILER CREATED TEMPORARIES.                          GPO         37    A
GPO      *           PASS 5  IS THE ASSEMBLY PASS.                        GPO         38    A
GPO                                                                       GPO         39    A
GPO      **        DEBUGGING SNAP LABELS -                                GPO         40    A
GPO      *                                                                GPO         41    A
GPO      *         TRACER (GPO,FUD,RDD,MII,FII,MIP,FXI,MIE,SHB,DIF,EIE,MPB)  GPO      42    A
GPO                SPACE  3                                               GPO         43    A
GPO       OF.MP    EQU    3          *MATERIALIZE* BIT IN *FLG* FIELD OF WORD B  GPO  44         I
         -CCGNC
GPO      *                           OF LOOP GENERATED LABELS ( *XX* ) .  GPO        45         I
         -CCGNC
GPO                                                                       GPO         46         I
         -CCGNC

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                LIST   -R,X                                                  GPO          47        I
          -CCGNC
GPO                                                                             GPO          48        I
          -CCGNC
GPO                TABLES BLK,TXT,GST,UDI,TET,PIT                               GPO          49        I
          -CCG1    -CCGNC
GPO                TABLES BST,IIT,MLT,MOD,RND,PSI,RXI                           GPO          50        I
          -CCG1    -CCGNC
GPO                ECHO 1,Z=(BLK,BST,GST,IIT,MLT,MOD,PIT,PSI,RND,RXI,TET,TXT,UDI)  CCG1      20        I
          -CCGNC
GPO                EXT    O.Z,L.Z                                              CCG1         21        I
          -CCGNC
GPO                                                                             GPO          51        I
          -CCGNC
GPO                LIST   -R,X                                                  CCGNC       523    A
GPO                                                                             CCGNC       524    A
GPO                TABLES (BLK,BST,GST,IIT,MLT,MOD,PIT,PSI,RND,RXI,TET,TXT,UDI)  CCGNC      525    A
GPO                TABLES (CFT,GLT)                                             CCGNC       526    A
GPO                EXT#   (TYIOM,STLDS,SUVA,ESRA,MX.BLN,PUNT,MOREFL,LM)         CCGNC       527    A
GPO       *CALL    CFTBLD                                                       GPO          52    A
GPO       *CALL    IOTBLD                                                       GPO          53    A
GPO       *CALL    UDTBLD                                                       GPO          54    A
GPO       *CALL    GPOCOM                                                       GPO          55    A
GPO        FLAGS   TITLE  VARIABLES                                             GPO          56    A
GPO       *        GLOBAL FLAGS                                                 GPO          57    A
GPO                                                                             GPO          58    A
GPO                ENTRY  DMASK                                                 GPO          59        I
          -FCC0509
GPO                ENTRY  DMASK,O.UDT                                           FCC0509       6        I
          -CCGNC
GPO        DMASK   BFMW   D,(STRS,USES)     FIELD CLEAR MASK                    GPO          60    A
GPO        DMASK#  EQENT  DMASK                                                 CCGNC       528    A
GPO        O$UDT   EQENT  O.UDT                                                 CCGNC       529    A
GPO                                                                             GPO          61    A
GPO       *                     PRE                                            GPO          62    A
GPO        N.GL0   ENTRY. =XN.GL      NEXT AVAIL GL-1 FOR MULT PRED LOOPS      GPO          63        I
          -CCG1
GPO        N.GL0   ENTRY. =XL.GLT     NEXT AVAIL GL-1 FOR MULTI PRED LOOPS     CCG1         22        I
          -CCGNC
GPO        N.GL0   BSSZ   1           NEXT AVAIL GL -1 FOR MULTI PRED LOOPS    CCGNC       530    A
GPO       *                     AUT                                            GPO          64    A
GPO                USE    /BVINFO/                                             GPO          65    A
GPO        VL      BSS    1           BIT VECTOR LENGTH                        GPO          66    A
GPO        N.VAR   ENTRY. 0           N. VARS IN *UDT*                         GPO          67        I
          -CCG7    -CCGNC
GPO        N.VAR   BSS    1           N. VARS IN *UDT*                         CCG7         93        I
          -CCGNC
GPO        BVL     ENTRY. 0           VL - 1                                   GPO          68        I
          -CCGNC
GPO        N.VAR   ENTRY. 0,#         N. VARS IN *UDT*                         CCGNC       531    A
GPO        BVL     ENTRY. 0,#         VL - 1                                   CCGNC       532    A
GPO        RCV     BSSZ   1           =1 IF *UDT* CONTAINS COMMON VARIABLES    GPO          69    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                 USE    0                                                    GPO          70     A
GPO       *                        DGS                                          GPO          71     A
GPO        CBN      ENTRY. 2           CURRENT BLOCK NUMBER                     GPO          72           I
          -CCG1
GPO        CG.CBN   ENTRY. 2           CURRENT FLOW BLOCK NUMBER                CCG1         23           I
          -CCGNC
GPO        CC$CBN   ENTRY. 2           CURRENT FLOW BLOCK NUMBER                CCGNC       533     A
GPO        NPB      ENTRY. 3           N.PROGRAM BLOCKS                         GPO          73           I
          -CCGNC
GPO        M.GRL    ENTRY. 0           MAX GRAPH LENGTH                         GPO          74           I
          -CCGNC
GPO        N.DG     ENTRY. 0           N.DERIVED GRAPHS ( MAX LOOP NESTING LEVEL )  GPO      75           I
          -CCGNC
GPO        GIC      ENTRY. 0           GRAPHS IN CORE FLAG , =0 IF INCORE       GPO          76           I
          -CCGNC
GPO        O.IST    ENTRY. =XPRE       FWA OF GRAPH STRUCTURE TABLES AFTER MOVE DO  GPO      77           I
          -CCG7     -CCGNC
GPO       #O.IST    ENTRY. =XF.BRDGE   FWA OF GRAPH STRUCTURE TABLES AFTER MOVE  #  CCG7       94           I
          -CCG9     -CCGNC
GPO        O.IST    ENTRY. =XF.BRDGE   FWA OF GRAPH STRUCTURE TABLES AFTER MOVE  CCG9      123           I
          -CCGNC
GPO        M.LP     ENTRY. 0           MAX N.LOOPS IN A GRAPH                   FCC0392      12           I
          -CCGNC
GPO                                                                             GPO          78           I
          -CCGNC
GPO       *                        GPO                                          GPO          79           I
          -CCGNC
GPO        M.BBL    ENTRY. 60B         MAX BASIC BLOCK LENGTH                   GPO          80           I
          -CCGNC
GPO                                                                             GPO          81           I
          -FCC0392  -CCGNC
GPO        M.EBL    ENTRY. 0           MAX EXTENDED BLOCK LENGTH                GPO          82           I
          -CCGNC
GPO        N.GT     ENTRY. 0           N. GLOBAL TEMPS ( IT."S )                GPO          83           I
          -CCGNC
GPO        O.BIT    ENTRY. O.UDB       FWA OF BLOCK INDEX TABLE                 GPO          84           I
          -CCGNC
GPO        L.BIT    ENTRY. 8           LENGTH                                   GPO          85           I
          -CCGNC
GPO        O.BVT    ENTRY. 0           FWA OF BIT VECTOR TABLE                  GPO          86           I
          -CCGNC
GPO        L.BVT    ENTRY. 0           LENGTH                                   GPO          87           I
          -CCGNC
GPO        L.SBV    BSS    1           LENGTH OF SPECIAL BIT VECTORS            GPO          88           I
          -CCGNC
GPO        L.PTXT   ENTRY. 0           LENGTH OF *FTNOPT* AT START OF *GPO*     GPO          89           I
          -CCGNC
GPO        O.LCT    ENTRY. 0           FWA OF LABEL CHANGE TABLE                FCC1319      18           I
          -CCGNC
GPO        L.LCT    ENTRY. 0           LENGTH  ( SET IN *CFA* )                 FCC1319      19           I
          -CCGNC
GPO       *                        PCC                                          GPO          90           I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
          -CCGNC
GPO        O.SEQ    ENTRY. 0           FWA OF BLOCK BEING PROCESSED                    GPO          91          I
          -CCGNC
GPO        L.SEQ    ENTRY. 0           LEN OF BLOCK BEING PROCESSED                    GPO          92          I
          -CCGNC
GPO        BN       ENTRY. 0           *BIT* INDEX OF BLOCK NUMBER                     GPO          93          I
          -CCGNC
GPO        BSI      ENTRY. 0           *BST* INDEX                                     GPO          94          I
          -CCGNC
GPO        BSW      ENTRY. 0           WORD 2 OF *BST* FOR CURRENT BLOCK               GPO          95          I
          -CCGNC
GPO       *                      SHB                                                   GPO          96          I
          -CCGNC
GPO        L.HB     ENTRY.             LENGTH OF HOLDING BLOCK                         GPO          97          I
          -CCGNC
GPO        NPB      ENTRY. 3,#         N. PROGRAM BLOCKS                               CCGNC       534     A
GPO        M.GRL    ENTRY. 0,#         MAX GRAPH LENGTH                                CCGNC       535     A
GPO        N.DG     ENTRY. 0,#         N.DERIVED GRAPHS ( MAX LOOP NESTING LEVEL )     CCGNC       536     A
GPO        GIC      ENTRY. 0,#         GRAPHS IN CORE FLAG , =0 IF INCORE              CCGNC       537     A
GPO        O.IST    ENTRY. =XF$BRDGE,# FWA OF GRAPH STRUCTURE TBLS AFTER MOVE DOWN     CCGNC       538     A
GPO        M.LP     ENTRY. 0,#         MAX  N.LOOPS IN A GRAPH                         CCGNC       539     A
GPO        N.FTH    ENTRY. 0,#         N. FTH LOOPS                                    CCGA021      28     A
GPO       *                      GPO                                                   CCGNC       540     A
GPO        M.BBL    ENTRY. 60D,#       MAX BASIC BLOCK LENGTH                          CCGNC       541     A
GPO        M.EBL    ENTRY. 0,#         MAX EXTENDED BLOCK LENGTH                       CCGNC       542     A
GPO        N.GT     ENTRY. 0,#         N. GLOBAL TEMPS                                 CCGNC       543     A
GPO        O$BIT    ENTRY. O.UDB       FWA OF BLOCK INDEX TABLE                        CCGNC       544     A
GPO        L$BIT    ENTRY. 8           LEN                                             CCGNC       545     A
GPO        O$BVT    ENTRY. 0           FWA OF BIT VECTOR TABLE                         CCGNC       546     A
GPO        L$BVT    ENTRY. 0           LEN                                             CCGNC       547     A
GPO        L.SBV    BSS    1         LEN OF SPECIAL BIT VECTORS                        CCGNC       548     A
GPO        L.PTXT   ENTRY. 0,#         LEN OF *FTNOPT* AT START OF *GPO*               CCGNC       549     A
GPO        O$LCT    ENTRY. 0           FWA OF LABEL CHANGE TABLE                       CCGNC       550         I
          -CCG18
GPO        L$LCT    ENTRY. 0           LEN   ( SET IN *CFA* )                          CCGNC       551         I
          -CCG18
GPO        O.LCT    ENTRY. ,#          FWA OF LABEL CHANGE TABLE                       CCG18        85     A
GPO        L.LCT    ENTRY. ,#          LEN  ( SET IN CFA )                             CCG18        86     A
GPO        O.ELT    ENTRY. 0,#         ORIGIN OF ELIMINATED LABEL DEF TABLE            CCGA021      29     A
GPO        L.ELT    ENTRY. 0,#         LENGTH  24/,18/IH PROG LAB,18/H OF GL           CCGA021      30     A
GPO       *                      PCC                                                   CCGNC       552     A
GPO        O$SEQ    ENTRY. 0           FWA OF BLOCK BEGIN PROCESSED                    CCGNC       553     A
GPO        L$SEQ    ENTRY. 0           LEN                                             CCGNC       554     A
GPO        BN       ENTRY. 0,#         *BIT* INDEX OF BLOCK NUMBER                     CCGNC       555     A
GPO        BSI      ENTRY. 0,#         *BSI* INDEX                                     CCGNC       556     A
GPO        BSW      ENTRY. 0,#         WORD 2 OF *BST* ENTRY OF CURRENT BLOCK          CCGNC       557     A
GPO       *                      SHB                                                   CCGNC       558     A
GPO        L.HB     ENTRY. 0,#         LEN OF HOLDING BLOCK                            CCGNC       559     A
GPO                                                                                    CCGNC       560     A
GPO                 ECHO   2,Z=(BIT,BVT,LCT,SEQ)                                       CCGNC       561         I
          -CCG18
GPO                 ECHO   2,Z=(BIT,BVT,SEQ)                                           CCG18        87     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

GPO        O.Z      EQU    O$Z                                                  CCGNC        562     A
GPO        L.Z      EQU    L$Z                                                  CCGNC        563     A
GPO                                                                             GPO           98     A
GPO       *         INTERVAL STRUCTURE TABLE POINTERS                            GPO           99     A
GPO                                                                             GPO          100     A
GPO        O.BTT    BSS    1           FWA OF BLOCK TRANSITION TABLE             GPO          101     A
GPO        O.HNT    BSS    1           FWA HEADER NODE TABLE                     FCC0392       13     A
GPO        L.HNT    BSS    1           LWA+1 OF *HNT*                            FCC0392       14     A
GPO       *                       IGP                                           GPO          102     A
GPO        GP       VFD    42/,18/=XPRE      GRAPH POINTER                       GPO          103           I
          -CCG7
GPO        GP       VFD    42/0,18/=XF.BRDGE GRAPH POINTER                       CCG7          95           I
          -CCG11
GPO        GP       VFD    42/,18/=XF$BRDGE  GRAPH POINTERS                      CCG11        113     A
GPO        EIT      BSS    1           EDGE INDEX TABLE ADDRESS                  GPO          104     A
GPO        N.LP     BSS    1           N. UNPROCESSED LOOPS IN CURRENT GRAPH     GPO          105     A
GPO        IBA      VFD    42/,18/COLA+1,60/1  INTERVAL BASE ADDR ( HEADER NODE ) GPO          106     A
GPO        NN       EQU    IBA+1       N. NODES IN INTERVAL                      GPO          107     A
GPO                                                                             GPO          108     A
GPO        BBV      CON    0           BUILD BIT VECTOR FLAG FOR *FUD*           GPO          109     A
GPO       *                           =0 IF TO FORM BIT VECTORS.                GPO          110     A
GPO        FLAGS    EJECT                                                        GPO          111     A
GPO        NBIP     ENTRY. 0           "0 IF LOOP CONTAINS NON-BASE *IP*S       FCC0514       21           I
          -CCGNC
GPO        NBIP     ENTRY. 0,#         "0 IF LOOP CONTAINS NON-BASE *IP*S       CCGNC        564     A
GPO        PSLC     ENTRY. 0           =0 IF PROCESSING STRAIGHT LINE CODE      FCC0514       22           I
          -FCC0675
GPO                                                                             FCC0514       23     A
GPO       *         LOCAL VARIABLES                                              GPO          112     A
GPO                                                                             GPO          113     A
GPO        HBI      BSS    1           BN OF HOLDING BLOCK                       GPO          114           I
          -FCC0675
GPO        HBI      ENTRY. 0           BN OF HOLDING BLOCK, 0 IF OUTSIDE ALL LPS FCC0675        9           I
          -CCGNC
GPO        HBI      ENTRY. 0,#         BN OF HOLDING BLOCK, 0 IF OUTSIDE ALL LPS CCGNC        565     A
GPO                 BSS    1           R1 OF HEADER, 12/P(OC.BOS),18/LINE,30/BN  GPO          115     A
GPO        MRA.X    BSS    1           *MRA* NOT REDUCED BY EXT REFS IN CASE ALL FCC1254        7     A
GPO       *                           ARE MOVED OUT OF THE LOOP AS INVARIANT    FCC1254        8     A
GPO        UXR      BSS    2           =1 IF EXT REFS IN BLOCK / REGION          FCC0448       31     A
GPO        FTH      BSSZ   1           < 0 IF FALL THROUGH TO HEADER LOOP        FCC0408        9     A
GPO                                                                             GPO          116     A
GPO                 USE    /TABLES/                                             GPO          117           I
          -CCG1
GPO                 USE    /CCG.SCR/                                            CCG1          24           I
          -CCG18
GPO                 USE    /CCGSCR/                                             CCG18         88     A
GPO                 BSS    100B        SCRATCH FOR *BDT* , ETC                  GPO          118           I
          -CCG16
GPO        CCG.SCR  BSS    100B        SCRATCH FOR BDT, ETC                     CCG16        200     A
GPO                                                                             GPO          119     A
GPO       *                       IRP                                           GPO          120     A
GPO        LGL      BSS    1           LOOP GENERATED LABEL, GLB = GEN LAB,      GPO          121           I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

          -FCC0909
GPO       *                          LL = LOOP LAB,  42/GLB-LL,18/LL                GPO          122          I
          -FCC0909
GPO        NTP      BSS    1          N. TERMINAL POLYNOMIALS                        GPO          123     A
GPO        PIC      BSS    1          POLYNOMIAL INCREMENT FORMULA LIST POINTER      GPO          124     A
GPO        TTI      BSS    1          *TET* INDEX FOR CURRENT BLOCK ( *UPB* )        GPO          125     A
GPO                                                                                 GPO          126     A
GPO       *                      FUD                                                GPO          127     A
GPO        UXR      BSS    2          =1 IF EXT REFS IN BLOCK / REGION               GPO          128          I
          -FCC0448
GPO        BVI      BSS    1          BIT VECTOR INDEX                               GPO          129     A
GPO                                                                                 GPO          130     A
GPO       *                      CHB                                                GPO          131     A
GPO        VI2      BSS    2          *BVT* INDICES OF THE SECOND AND FIRST BLOCK    GPO          132     A
GPO        BSIS     BSS    1          30/BST OF PRED,30/BSI OF CURRENT BLOCK         GPO          133     A
GPO                                                                                 GPO          134     A
GPO       *                      MII                                                GPO          135     A
GPO        AN       BSS    1          1S59 IF BLOCK IS AN ARTICULATION NODE          GPO          136     A
GPO        UBD      BSS    1          BLOCK *UBD* BIT VECTOR ADDRESS                 GPO          137     A
GPO        IJP      BSS    2          INVARIANT JUMP ENCOUNTERED FLAG                GPO          138     A
GPO        UIC      EQU    IJP+1      UNSAFE INVARIANT CODE ENCOUNTERED              GPO          139     A
GPO                                                                                 GPO          140     A
GPO       *                      FII                                                GPO          141     A
GPO        IIL      BSS    1          INITIAL *IIT* LENGTH                           GPO          142     A
GPO                                                                                 GPO          143     A
GPO       *                      MIE                                                GPO          144     A
GPO        MINC     BSS    1          MIN COST *IP* VALUES THAT WE WILL MOVE         GPO          145     A
GPO        IC       BSS    1          INSTRUCTION CHAIN                              GPO          146     A
GPO        ISV      BSS    1          SAVED VALUE OF I ( MULTIPLE DEF"S )            FCC1254        9     A
GPO        TBB      BSS    1          FWA OF HOLDING BLOCK                           GPO          147     A
GPO                                                                                 GPO          148     A
GPO       *                      CIF                                                GPO          149     A
GPO        CAS      BSS    1          *CA* OF TERMINAL POLYNOMIAL                    GPO          150     A
GPO        TSC      BSS    1          TEMP STORE CHAIN POINTER                       GPO          151     A
GPO        CNI      EQU    TSC        CLASS NUMBER INCREMENT                         FCC0514       24     A
GPO                                                                                 GPO          152     A
GPO       *                      FIM                                                GPO          153     A
GPO        SC       BSS    1          *TXT* INDEX OF *ST*                            GPO          154     A
GPO        J        BSS    1          *UDI* INDEX OF *ST*                            GPO          155     A
GPO        EV       BSS    1          EXPRESSION VALUE                               GPO          156     A
GPO        PI       BSS    2          POLYNOMIAL INCREMENT FORMULA INDEX             GPO          157     A
GPO        ITB      BSS    1          L.IIT WHEN START TO PROCESS A *ST*             GPO          158     A
GPO        .LL      SET    CCG.SCR+260B-*                                            CCG16        201          I
          -CCG18
GPO                 BSS    .LL*HC.2OC/HC.2OC ENSURE FIRST BLK SIZE DEF IS LARGEST    CCG16        202          I
          -CCG18
GPO                 BSS    CCG.SCR+260B-*    ENSURE FIRST BLOCK SIZE IS LARGEST      CCG18         89     A
GPO                 USE    0                                                         GPO          159     A
GPO        STRUCT   TITLE  TABLE FORMATS                                            GPO          160     A
GPO       *         BS. - BLOCK STATUS INFORMATION WORD ( *MII* TO *UPB* )          GPO          161     A
GPO       *         INITIALLY THE *BS* WORD CONTAINS THE *IM* WORD OF THE BLOCK     GPO          162     A
GPO                                                                                 GPO          163     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                DESCRIBE BS.,60                                          GPO          164     A
GPO                DESCRIBE 2                                               GPO          165     A
GPO       NIS      DEFINE 10          P(N.INCREMENT STORES/N.POLYNOMIAL INCRS)  GPO     166     A
GPO       BM       DEFINE 1           BLOCK MODIFIED                        GPO          167     A
GPO       NTET     DEFINE 11          N.TET"S IN BLOCK                      GPO          168     A
GPO       STC      DEFINE 18          *RD* STORE CHAIN                      GPO          169     A
GPO       IIC      DEFINE 18          INVARIANT / INTEGER POLYNOMIAL CHAIN  GPO          170     A
GPO       LINK     SPACE  3,14                                              GPO          171     A
GPO      *         I. - LINK INFORMATION WORD FORMAT DURING *GPO*           GPO          172     A
GPO      *         THE LINK WORD IS IN I. FORMAT FROM *MII* TO *UPB*        GPO          173     A
GPO                                                                         GPO          174     A
GPO                DESCRIBE I.,60                                           GPO          175     A
GPO       INV      DEFINE 1           INVARIANT INSTRUCTION                 GPO          176     A
GPO       M        DEFINE 1           MARKED ( IN *DIF* )                   GPO          177     A
GPO       TI       DEFINE 1           TERMINAL EXPRESSION                   GPO          178     A
GPO                                                                         GPO          179     A
GPO       IA       DEFINE 1           INTEGER ADD                           GPO          180     A
GPO       IP       DEFINE 1           INTEGER POLYNOMIAL                    GPO          181     A
GPO                DEFINE 1                                                 GPO          182         I
          -FCC0717
GPO       AIP      DEFINE 1           ABNORMAL IP, TERMINAL IP USES OLD VALUE OF  FCC0717  6     A
GPO      *                            RD VAR & IS PAST AN INCR OF IT.       FCC0717        7     A
GPO       COST     DEFINE 9           *COST* OF AN *IP*                     GPO          183     A
GPO       USES     DEFINE 9           COUNT OF REMOVED USES                 GPO          184     A
GPO       IIC      DEFINE 18          INV INST CHAIN, ETC                   GPO          185     A
GPO       HBP      DEFINE 18          HOLDING BLOCK POINTER                 GPO          186     A
GPO                                                                         GPO          187     A
GPO       LDI      DEQU   HBP         *UDI* INDEX OF *LD* IF INST IS *IA* / *IP*  GPO    188     A
GPO       STC      DEQU   IIC         STORE CHAIN OF RECURSIVELY DEFINED VARS  GPO      189     A
GPO                                                                         GPO          190     A
GPO       C1       DEQU   COST,1      LOW BIT OF FIELD                      GPO          191     A
GPO                                                                         GPO          192     A
GPO       IAIP     DEQU   IP,2        IA & IP BITS                          GPO          193     A
GPO       HNT      SPACE  3                                                 FCC0392       15     A
GPO      **        H. - HEADER NODE TO *HB* TRANSITION TABLE ( IGP, FXI )   FCC0392       16     A
GPO                                                                         FCC0392       17     A
GPO                DESCRIBE H.,60     HNW(NIP,HBI,HNI)                      FCC0392       18     A
GPO                DEFINE 6                                                 FCC0392       19     A
GPO       NIP      DEFINE 18          N.INTERNAL PRED OF INTERVAL HEADER    FCC0392       20     A
GPO       HBI      DEFINE 18          BIT INDEX OF HB HEADER                FCC0392       21     A
GPO       HNI      DEFINE 18          BIT INDEX OF HEADER OF INTERVAL       FCC0392       22     A
GPO                                                                         GPO          194     A
GPO                LIST   R,-X                                              GPO          195     A
GPO       AFB      TITLE  SUBROUTINES                                       GPO          196     A
GPO      **        AFB - ALLOCATE FIXED BLOCK AT LOW MEMORY                 GPO          197     A
GPO      *                                                                  GPO          198     A
GPO      *         ENTRY  (X1) = LEN , OF BLOCK                             GPO          199     A
GPO      *         EXIT   (X6) = FWA OF BLOCK                               GPO          200     A
GPO      *                (X3) = (X7) = LEN                                 GPO          201     A
GPO                                                                         GPO          202     A
GPO       AFB      ROUTINE                                                  GPO          203     A
GPO                ALLOC  BLK,X1      ALLOC( BLK , LEN )                    GPO          204     A

          0        1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                IX6    X2+X3                                                GPO          205     A
GPO                SA6    A2          O.BLK = O.BLK + LEN                      GPO          206     A
GPO                SA6    LM          LM = O.BLK                               GPO          207     A
GPO                MX7    0                                                    GPO          208     A
GPO                SA7    A3          L.BLK = 0                                GPO          209     A
GPO                BX6    X2                                                   GPO          210     A
GPO                BX7    X3                                                   GPO          211     A
GPO                EQ     AFB                                                  GPO          212     A
GPO        STO     SPACE  3                                                    GPO          213     A
GPO       **       STO - SET MANAGED TABLES ORIGIN ( LOW MEMORY LIMIT )        GPO          214     A
GPO       *                                                                    GPO          215     A
GPO       *        ENTRY  (X7) = NLM , NEW LOW MEMORY ADDRESS                  GPO          216     A
GPO       *        EXIT   LM = O.TXT = NLM , L.TXT = 0                         GPO          217     A
GPO                                                                            GPO          218     A
GPO        STO     ROUTINE                                                     GPO          219     A
GPO                SA5    =XLM                                                 GPO          220     A
GPO                CALL   AMU=        ACCUMULATE MEMORY USED                   GPO          221         I
          -CCG7
GPO                IX6    X6+X5       NEW = LM + MU                            GPO          222         I
          -CCG7
GPO                SA4    =XP2.FL                                              GPO          223         I
          -CCG7
GPO                IX0    X6-X4                                                GPO          224         I
          -CCG7
GPO                MI     X0,STO1     P2.FL = MAX( NEW , P2.FL )               GPO          225         I
          -CCG7
GPO                SA6    A4+                                                  GPO          226         I
          -CCG7
GPO                                                                            GPO          227         I
          -CCG7
GPO        STO1    SA7    A5          LM = NLM                                 GPO          228         I
          -CCG7
GPO                SA7    A5+         LM = NLM                                 CCG7          96     A
GPO                SA7    =XO.TXT     O.TXT = LM                               GPO          229     A
GPO                MX6    0                                                    GPO          230     A
GPO                SA6    =XL.TXT     L.TXT = 0                                GPO          231     A
GPO                SA6    =XMU        MU = 0                                   GPO          232         I
          -CCG7
GPO                EQ     STO                                                  GPO          233     A
GPO        WTB     SPACE  2                                                    GPO          234     A
GPO       **       WTB - WRITE *TXT* BLOCK BACK TO *BLK*                       GPO          235     A
GPO                                                                            GPO          236     A
GPO        WTB     ROUTINE                                                     GPO          237     A
GPO                SA1    O.TXT                                                GPO          238     A
GPO                SA2    L.TXT                                                GPO          239     A
GPO                SA3    X1+B1       R2 = [O.TXT+1]                           GPO          240     A
GPO                MX4    -R2.TXTLL                                            GPO          241     A
GPO                LX3    -R2.TXTLP                                            GPO          242     A
GPO                BX5    X4*X3                                                GPO          243     A
GPO                IX6    X5+X2       TXTL[R2] = L.TXT                         GPO          244     A
GPO                LX6    R2.TXTLP                                             GPO          245     A
GPO                SA6    A3                                                   GPO          246     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                CALL   WMB         REWRITE THE BLOCK                          GPO          247          I
          -CCGNC
GPO                CALL   WMB#        REWRITE THE BLOCK                          CCGNC        566     A
GPO                SX6    4                                                      GPO          248     A
GPO                SA6    L.TXT       L.TXT = 4                                  GPO          249     A
GPO                EQ     WTB                                                    GPO          250     A
GPO       PCC      TITLE  PCC - PROCESS COROUTINE CALL                           GPO          251     A
GPO       **       PCC - PROCESS COROUTINE CALL                                  GPO          252     A
GPO       *               CALL *RTN* FOR ALL BLOCKS IN THE *BST* BUT THE FIRST.  GPO          253     A
GPO       *                                                                     GPO          254     A
GPO       *        ENTRY  (X7) = RTN , SUBROUTINE NAME                           GPO          255     A
GPO       *               (X5) = BSFM , BLOCK STATUS FIELD MASK                  GPO          256     A
GPO       *               IF = 0, THEN CALL *RTN* FOR ALL BLOCKS.                GPO          257     A
GPO       *                                                                     GPO          258     A
GPO       *        THE CALLED COROUTINE IS ENTERED WITH -                        GPO          259     A
GPO       *               (X0) = R-NUMBER EXTRACTOR MASK                         GPO          260     A
GPO       *               (X4) = BSW , SECOND WORD OF *BST* ENTRY                GPO          261     A
GPO       *               (X5) = O.SEQ = FWA OF BLOCK                            GPO          262     A
GPO       *               (L.SEQ) = BLOCK LENGTH                                 GPO          263     A
GPO       *               (BN) = *BIT* INDEX OF BLOCK                            GPO          264     A
GPO       PCC      SPACE  2                                                      GPO          265     A
GPO       PCCA     SA5    O.SEQ                                                  GPO          266     A
GPO                RJ     0                                                      GPO          267     A
GPO                BSS    1           BSFM SAVE WORD                             GPO          268     A
GPO                                                                              GPO          269     A
GPO       PCC      ENTRY. **                                                     GPO          270          I
          -CCGNC
GPO       PCC      ENTRY. **,#                                                   CCGNC        567     A
GPO                SA1    PCCA                                                   GPO          271     A
GPO                LX6    X5                                                     GPO          272     A
GPO                BX7    X1+X7                                                  GPO          273     A
GPO                SA6    A1+B1                                                  GPO          274     A
GPO                SA7    PCC5        [PCC5] = *  RJ RTN *                       GPO          275     A
GPO                SX6    B1+B1       BSI = 2                                    GPO          276     A
GPO                RJ     *+1         VOID INSTRUCTION STACK                     CCGA090        8     A
GPO       +        EQ     *                                                      CCGA090        9     A
GPO                                                                              GPO          277     A
GPO       PCC1     SA5    O.BST                                                  GPO          278     A
GPO                IX7    X5+X6                                                  GPO          279     A
GPO                SA1    X7          BAW = BST(BSI)                             GPO          280     A
GPO                SA2    X7+1        BSW = BST(BSI+1)                           GPO          281     A
GPO                SA3    PCCA+1                                                 GPO          282     A
GPO                ZR     X1,PCC      IF BAW = 0    */ END OF *BST*              GPO          283     A
GPO                ZR     X3,PCC2     IF BSFM = 0   */ NO FIELD MASK             GPO          284     A
GPO                BX4    X3*X2                                                  GPO          285     A
GPO                ZR     X4,PCC6     IF BSFM[BSW] = 0  */ FIELDS NOT SET        GPO          286     A
GPO                                                                              GPO          287     A
GPO       PCC2     BX7    X2                                                     GPO          288     A
GPO                SA6    BSI                                                    GPO          289     A
GPO                LX1    -BA.BIP                                                GPO          290     A
GPO                SX6    X1                                                     GPO          291     A
GPO                SA7    BSW                                                    GPO          292     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO              LX1    59-BA.BLKP+BA.BIP                                GPO        293    A
GPO              SA6    BN          BN = BI[BAW]                         GPO        294    A
GPO              MI     X1,PCC3     IF BLK[BAW]   */ BLOCK IS IN *BLK*   GPO        295    A
GPO                                                                      GPO        296    A
GPO              CALL   RTB         READ IT INTO *BLK*                   GPO        297         I
         -CCGNC
GPO              CALL   RTB#        READ IT INTO *BLK*                   CCGNC      568    A
GPO              EQ     PCC4                                             GPO        298    A
GPO                                                                      GPO        299    A
GPO       PCC3   LX1    1+BA.BLKP-BA.FWAP                                GPO        300    A
GPO              SX6    X1                                               GPO        301    A
GPO              SA6    O.SEQ       O.SEQ = FWA[BAW]                     GPO        302    A
GPO              LX1    BA.FWAP-BA.LENP                                  GPO        303    A
GPO              SX7    X1+                                              GPO        304    A
GPO              SA7    L.SEQ                                            GPO        305    A
GPO       PCC    SPACE  1,10                                             GPO        306    A
GPO      *       SETUP REGISTERS AND CALL COROUTINE                      GPO        307    A
GPO                                                                      GPO        308    A
GPO       PCC4   MX0    -R1.RIL+1                                        GPO        309         I
         -CCG11
GPO              SA4    BSW                                              GPO        310         I
         -CCG11
GPO              LX0    1                                                GPO        311         I
         -CCG11
GPO       PCC4   SA4    BSW                                              CCG11      114    A
GPO              SX0    RN.MASK                                          CCG11      115    A
GPO       PCC5   SA5    O.SEQ                                            GPO        312    A
GPO              RJ     *           CALL COROUTINE                       GPO        313    A
GPO              SA1    O.BST                                            GPO        314    A
GPO              SA2    BSI                                              GPO        315    A
GPO              SA3    BSW                                              GPO        316    A
GPO              IX5    X1+X2                                            GPO        317    A
GPO              BX7    X3                                               GPO        318    A
GPO              SX6    X2+                                              GPO        319    A
GPO              SA7    X5+1        BST(BSI+1) = BSW                     GPO        320    A
GPO                                                                      GPO        321    A
GPO       PCC6   SX6    X6+2        BSI = BSI + 2                        GPO        322    A
GPO              EQ     PCC1                                             GPO        323    A
GPO       SRI    SPACE  3,14        SRI                                  GPO        324    A
GPO      **      SRI/SII - STORE RLIST/INCREMENT INSTRUCTION             GPO        325    A
GPO      *                                                               GPO        326    A
GPO      *       ENTRY  (B2,X6,X7) = OC, R1, R2                          GPO        327    A
GPO      *              (B6,B7) = TI, TB ( BLOCK STORE INDEX , BLOCK FWA ) GPO      328    A
GPO      *                                                               GPO        329    A
GPO      *       EXIT   (X7) = 0 , TI = TI + 4                           GPO        330    A
GPO      *                                                               GPO        331    A
GPO      *       USES   A - 4, 6, 7   X - 4, 6, 7                        GPO        332    A
GPO                                                                      GPO        333    A
GPO       SRI    ENTRY. **                                               GPO        334         I
         -CCGNC
GPO              SA4    =XF.RDT+B2                                       GPO        335         I
         -CCGNC

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO       SRI     ENTRY. **,#                                                  CCGNC      569    A
GPO               SA4    F.RDT+B2                                              CCGNC      570    A
GPO               PX6    B2,X6                                                 GPO        336    A
GPO               SA6    B7+B6       TA = TB + TI;  [TA] = PACK(OC,R1)         GPO        337    A
GPO               SB6    B6+4        TI = TI + 4                               GPO        338    A
GPO               SA7    A6+B1       [TA+1] = R2                               GPO        339    A
GPO               BX6    X4                                                    GPO        340    A
GPO               SA6    A7+B1       [TA+2] = RDT(OC)                          GPO        341    A
GPO               SX7    0                                                     GPO        342    A
GPO               SA7    A6+B1       [TA+3] = 0                                GPO        343    A
GPO               EQ     SRI                                                   GPO        344    A
GPO                                                                            GPO        345    A
GPO       SII     ROUTINE                                                      GPO        346    A
GPO               SA4    F.RDT+B2                                              GPO        347    A
GPO               PX6    B2,X6                                                 GPO        348    A
GPO               SA6    B7+B6       TA = TB + TI;  [TA] = PACK(OC,R1)         GPO        349    A
GPO               SB6    B6+4        TI = TI + 4                               GPO        350    A
GPO               SX6    B1                                                    GPO        351    A
GPO               SA7    A6+B1       [TA+1] = R2                               GPO        352    A
GPO               LX6    D.INCP                                                GPO        353    A
GPO               MX7    0                                                     GPO        354    A
GPO               BX6    X6+X4                                                 GPO        355    A
GPO               SA6    A7+B1       [TA+2] = RDT(OC) ! *INC*                  GPO        356    A
GPO               SA7    A6+B1       [TA+3] = 0                                GPO        357    A
GPO               EQ     SII                                                   GPO        358    A
GPO       SMI     SPACE  3,14                                                  FCC0656      9    A
GPO      **       SMI - STORE MEMORY REFERENCE INSTRUCTION, SET FP/LVL BITS    FCC0656     10    A
GPO      *                IN DESCRIPTOR.                                      FCC0656     11    A
GPO      *                                                                    FCC0656     12    A
GPO      *        ENTRY/EXIT - SAME AS *SRI/SII*                              FCC0656     13    A
GPO                                                                            FCC0656     14    A
GPO       SMI     ROUTINE                                                      FCC0656     15    A
GPO               SA4    F.RDT+B2    D = RDT(OC)                               FCC0656     16    A
GPO               PX6    B2,X6                                                 FCC0656     17    A
GPO               SA6    B7+B6       TA = TB + TI;  [TA] = PACK(OC,R1)         FCC0656     18    A
GPO               SA7    A6+B1       [TA+1] = R2                               FCC0656     19    A
GPO               BX6    X4                                                    FCC0656     20    A
GPO               LX7    1                                                     FCC0656     21          I
         -CCG1
GPO               SA4    O.SYM                                                 FCC0656     22          I
         -CCG1
GPO               SB6    X7+B1                                                 FCC0656     23          I
         -CCG1
GPO               SB2    X4                                                    FCC0656     24          I
         -CCG1
GPO               SA4    B2-B6       WORDB = [O.SYM-2*IH[R2]-1]                FCC0656     25          I
         -CCG1
GPO               LX4    -P.FPB                                                FCC0656     26          I
         -CCG1
GPO               SA4    =XO.SYM                                               CCG1        25    A
GPO               SB6    X4+B1                                                 CCG1        26    A
GPO               SB2    B6+X7                                                 CCG1        27    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                LX7    1                                                  CCG1          28     A
GPO                SA4    B2+X7       WORDB = SYM(3*IH+1)                    CCG1          29     A
GPO                LX4    -WB.FPP                                            CCG1          30     A
GPO                SX7    B1                                                 FCC0656       27     A
GPO                BX7    X7*X4                                              FCC0656       28     A
GPO                LX7    D.FPP       FP[D] = FP[WORDB]                      FCC0656       29     A
GPO                BX6    X7+X6                                              FCC0656       30     A
GPO                SX7    B1+B1                                              FCC0656       31     A
GPO                BX7    X7*X4                                              FCC0656       32     A
GPO                LX7    D.L2P-1                                            FCC0656       33     A
GPO                BX6    X7+X6       L2[D] = L2[WORDB]                      FCC0656       34     A
GPO                SA6    A7+B1       [TA+2] = D                             FCC0656       35     A
GPO                MX7    0                                                  FCC0656       36     A
GPO                SA7    A6+B1       [TA+3] = 0                             FCC0656       37     A
GPO                SB6    A7+B1                                              FCC0656       38     A
GPO                SB6    B6-B7       TI = TI + 4                            FCC0656       39     A
GPO                EQ     SMI                                                FCC0656       40     A
GPO        COL     TITLE  COL - CODE OPTIMIZABLE LOOP ( OPT=1 )              GPO          359     A
GPO       **       COL - CODE OPTIMIZABLE LOOP ( OPT=1 )                     GPO          360     A
GPO       *                                                                  GPO          361     A
GPO       *        PROCESS INNERMOST *DO* LOOPS THAT ARE SINGLE ENTRY/ SINGLE GPO          362     A
GPO       *        EXIT, AND HAVE NO EXTERNAL REFERENCES OR BACKWARDS BRANCHES. GPO        363     A
GPO       *        OPTIMIZATIONS PERFORMED ARE CODE MOTION, STRENGTH REDUCTION, GPO        364     A
GPO       *        THE ASSIGNMENT OF INTEGER VARIABLES TO B-REGISTERS AND     GPO          365     A
GPO       *        TEST REPLACEMENT.                                          GPO          366     A
GPO       *        SINCE WE DO NOT HAVE THE LIVE EXIT INFORMATION AVAILABLE, WE GPO        367     A
GPO       *        ASSUME THAT ALL VARIABLES DEFINED IN THE LOOP ARE LIVE ON  GPO          368     A
GPO       *        EXIT FROM IT.                                              GPO          369     A
GPO                                                                           GPO          370     A
GPO                USE    //                                                 GPO          371          I
          -CCG18
GPO        O.UDB   BSS    1B          FWA OF *USE/DEF* HASH, FWA OF WORKING STORA GPO     372          I
          -CCG18
GPO                USE    0                                                  GPO          373          I
          -CCG18
GPO                EXT    F.MEM                                              CCG18         90     A
GPO        O.UDB   EQU    F.MEM                                              CCG18         91     A
GPO                                                                           GPO          374     A
GPO        SPC     BSS    1           SAVED PARCEL COUNT                     GPO          375     A
GPO        SERF    BSS    1           SAVED VALUE OF HO$ER                   CCG28          1     A
GPO                                                                           GPO          376     A
GPO       *        INTERVAL INFORMATION                                       GPO          377     A
GPO                                                                           GPO          378     A
GPO        COLA    VFD    42/3,18/1   PB = 3 , NODE = 1                      GPO          379     A
GPO                VFD    6/77B,18/4,18/,18/2                                GPO          380     A
GPO                                                                           GPO          381     A
GPO        COL     ENTRY. **                                                 GPO          382          I
          -CCGNC
GPO        COL     ENTRY. **,#                                               CCGNC        571     A
GPO                QUAL   CODE                                               GPO          383     A
GPO                                                                           GPO          384     A
GPO                SA2    PARCEL                                             GPO          385          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

          -CCG7
GPO                SX6    X2+         SPC = PARCEL  */ SAVE PARCEL COUNT                GPO          386          I
          -CCG7    -CCG28
GPO                SA3    =XHO$ER                                                       CCG28          2     A
GPO                SX6    X2                                                            CCG28          3     A
GPO                BX7    X3                                                            CCG28          4     A
GPO                SA2    =XCG.PC                                                       CCG7          97          I
          -CCGNC
GPO                SA2    =XCC$PC                                                       CCGNC        572     A
GPO                SX6    X2+         SPC = CG.PC   */ SAVE PARCEL COUNTER              CCG7          98          I
          -CCGNC
GPO                SX6    X2+         SPC = CC$PC   */ SAVE PARCEL COUNTER              CCGNC        573     A
GPO                SA6    SPC                                                           GPO          387     A
GPO                SA7    A6+B1       SERF = HO$ER                                      CCG28          5     A
GPO                MX6    0                                                             CCG28          6     A
GPO                SA6    A3          HO$ER = 0     */ CLEAR WHILE CODEING LOOP         CCG28          7     A
GPO                CALL   AFT=        ACTIVATE *BLK*                                    GPO          388          I
          -CCGNC
GPO                CALL   AFT#        ACTIVATE *BLK*                                    CCGNC        574     A
GPO                ALLOC  BLK,100B    ALLOC( BLK , 100B )  */ ALLOC FOR *UDB*            GPO          389     A
GPO                SETZERO O.UDB,X1                                                     GPO          390     A
GPO                ALLOC  UDI,2       ALLOC( UDI , 2 )                                  GPO          391     A
GPO                MX6    0                                                             GPO          392     A
GPO                SA6    X2          UDI(0) = 0                                        GPO          393     A
GPO                SA6    X2+1        UDI(1) = 0                                        GPO          394     A
GPO                CALL   CMR         CHAIN MEM REFS TO *UDI*                           GPO          395          I
          -CCGNC
GPO                CALL   AUT         ADJUST USE TABLE                                  GPO          396          I
          -CCGNC
GPO                CALL   CMR#        CHAIN MEMREFS TO *UDI*                            CCGNC        575     A
GPO                CALL   AUT#        ADJUST USE TABLE                                  CCGNC        576     A
GPO                SX6    0                                                             GPO          397     A
GPO                SA6    L.BLK       L.BLK = 0                                         GPO          398     A
GPO                SX1    8                                                             GPO          399     A
GPO                RJ     AFB         AFB( 8 )      */ ALLOCATE FOR *BIT*               GPO          400     A
GPO                SETZERO O.UDB,8                                                      GPO          401     A
GPO                MX0    1                                                             GPO          402     A
GPO                LX0    BI.RBP                                                        GPO          403          I
          -FCC0641
GPO                LX0    1+BI.RBP                                                      FCC0641        8     A
GPO                SX1    B1                                                            GPO          404     A
GPO                LX1    BI.ILP                                                        GPO          405     A
GPO                BX6    X0+X1                                                         GPO          406     A
GPO                SA6    O.UDB+4     (RB,IL)BIT(4) = 1                                 GPO          407     A
GPO                EQ     IGO0        GO SETUP *UDT*                                    GPO          408     A
GPO                                                                                     GPO          409     A
GPO       COL1     CALL   IMPD        SETUP *BLK* , MOVE LOOP BODY TO BLK               GPO          410          I
          -CCGNC
GPO       COL1     CALL   IMPD#       SETUP *BLK* , MOVE LOOP BODY TO BLK               CCGNC        577     A
GPO                RJ     IRP         INITIALIZE REGION PROCESSING                      GPO          411     A
GPO                MX7    UD.MVL                                                        GPO          412     A
GPO                LX7    UD.MVL+UD.MVP                                                 GPO          413     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                 SA7    BCM         BCM = *BLOCK CLEAR BITS*                      GPO          414     A
GPO                 CLCM   FUD         FORM USE/DEF INFO                             GPO          415          I
          -CCGNC
GPO                 CLCM   FUD#        FORM USE/DEF INFO                             CCGNC        578     A
GPO                                                                                 GPO          416     A
GPO       *         SETUP SPECIAL BIT VECTORS AND *RXI* SO AS TO COMPENSATE FOR     GPO          417     A
GPO       *         THE LACK OF LIVE EXIT INFO. SET TO CODE POST STORE IN EPILOGUE  GPO          418     A
GPO                                                                                 GPO          419     A
GPO                 SA4    O.UDB+4                                                  GPO          420     A
GPO                 SA3    O.BVT                                                    GPO          421     A
GPO                 SA1    VL                                                       GPO          422     A
GPO                 SA2    DVA                                                      GPO          423     A
GPO                 LX4    -BI.BVIP                                                 GPO          424     A
GPO                 SB7    X4          VI = BVI[BIT(4)]                             GPO          425     A
GPO                 SB2    B0          I = 0                                        GPO          426     A
GPO                 SB3    X1                                                       GPO          427     A
GPO                 SB4    B3+B3                                                    GPO          428     A
GPO                 SA5    X3+B7       DV = BVT + VI                                GPO          429     A
GPO                 SA4    A5+B4       UV = DV + 2*VL                               GPO          430     A
GPO                 SX0    X1                                                       GPO          431     A
GPO        COL2     BX6    X5                                                       GPO          432     A
GPO                 BX7    X4+X5                                                    GPO          433     A
GPO                 SA6    X2+B2       DEF                                          GPO          434     A
GPO                 SA6    A6+B3       LU                                           GPO          435     A
GPO                 SA7    A6+B3       USE                                          GPO          436     A
GPO                 SA6    A7+B3       LX                                           GPO          437     A
GPO                 SA7    A6+B3       LE                                           GPO          438     A
GPO                 SA6    A7+B3       MD                                           GPO          439     A
GPO                 SB2    B2+B1       I = I + 1                                    GPO          440     A
GPO                 LT     B2,B3,COL2  IF I < VL                                    GPO          441     A
GPO                                                                                 GPO          442     A
GPO                 ALLOC  RXI,B2+3    ALLOC( RXI , VL+3 )                          GPO          443     A
GPO                 SX6    B0                                                       GPO          444     A
GPO                 MX7    1                                                        GPO          445     A
GPO                 LX7    1+E.SCRP                                                 GPO          446     A
GPO                 SA6    X2          RXI(0) = 0                                   GPO          447     A
GPO                 SA7    X2+B1       RXI(1) = E(SCR,0,0,0)                        GPO          448     A
GPO                 SA2    DVA                                                      GPO          449     A
GPO                 SB2    X0+B1                                                    GPO          450     A
GPO                 SA6    A7+B2       RXI(VL+2) = 0                                GPO          451     A
GPO                 MOVE   X0,X2,A7+B1   MOVE( VL , DVA , RXI(2) )                  GPO          452     A
GPO                 SX7    2                                                        GPO          453     A
GPO                 SA7    EPSI        EPSI = 2                                     GPO          454     A
GPO                                                                                 GPO          455     A
GPO                 CLCM   MII         MARK INVARIANT INSTRUCTIONS                  GPO          456     A
GPO                 RJ     IHB         INITIALIZE HOLDING BLOCK                     GPO          457     A
GPO                 PLUG   AT=GPO7,TO=COL4,VOID=NO                                  GPO          458     A
GPO                 EQ     GPO3A       GO FINISH OFF THE CODE MOTION PHASE          GPO          459          I
          -FCC0646
GPO                                                                                 GPO          460          I
          -FCC0646
GPO       *         CODE MOTION PERFORMED, EXPAND ANY LEVEL REFS IN THE LOOP        GPO          461          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
                -FCC0646
GPO             *         AND CALL *GRA* TO CODE IT , AND THE EPILOGUE POST STORES.          GPO        462        I
                -FCC0646
GPO                                                                                          GPO        463        I
                -FCC0646
GPO              COL4     SX7    2                                                           GPO        464        I
                -FCC0646
GPO                       SA7    BSI         BSI = 2                                         GPO        465        I
                -FCC0646
GPO                       CALL   RTB         GET THE LOOP BODY                               GPO        466        I
                -FCC0646
GPO                       SX5    2                                                           GPO        467        I
                -FCC0646
GPO                       SX6    1                                                           GPO        468        I
                -FCC0646
GPO                       CALL   ESR         EXPAND LEVEL REFS INSIDE THE LOOP               GPO        469        I
                -FCC0646
GPO                       CALL   WTB         REWRITE THE BLOCK BACK TO *BLK*                 GPO        470        I
                -FCC0646
GPO                       SX6    100                                                         GPO        471        I
                -FCC0646
GPO                       SA6    MAXW        MAXW = 100    */ INHIBIT X-REGISTER ASSIGNM     GPO        472        I
                -FCC0646
GPO                       CALL   GRA         CODE THE LOOP BODY                              GPO        473        I
                -FCC0646
GPO                       SX6    100                                                         FCC0646      8        I
                -CCG9
GPO                       SA6    MAXW        MAXW = 100    */ INHIBIT X-ASSIGNMENT IN *GRA*  FCC0646      9        I
                -CCG9
GPO                       EQ     GPO3A       FINISH OFF CODE MOTION & CODE THE LOOP          FCC0646     10     A
GPO                                                                                          GPO        474     A
GPO             *         NOW CONVERT THE *HB* TO CODE                                       GPO        475     A
GPO                                                                                          GPO        476     A
GPO                       SX6    0                                                           GPO        477        I
                -FCC0646
GPO              COL4     SX6    0                                                           FCC0646     11     A
GPO                       SA6    BSI         BSI = 0                                         GPO        478     A
GPO                       CALL   RTB                                                         GPO        479        I
                -CCGNC
GPO                       CALL   RTB#                                                        CCGNC      579     A
GPO                       SA1    =XL.ALS                                                     GPO        480        I
                -CCG7
GPO                       SA1    =XCG.LNL                                                    CCG7        99        I
                -CCGNC
GPO                       SA1    =XCC$LNL                                                    CCGNC      580     A
GPO                       SX5    3                                                           GPO        481     A
GPO                       SX1    X1-1                                                        GPO        482     A
GPO                       ZR     X1,COL5     IF L.ALS = 1  */ OUTSIDE OF ALL LOOPS            GPO        483        I
                -CCG7
GPO                       ZR     X1,COL5     IF CG.LNL = 1 */ OUTSIDE OF ALL LOOPS            CCG7       100        I
                -CCGNC
GPO                       ZR     X1,COL5     IF CG$LNL = 1 */ OUTSIDE OF ALL LOOPS            CCGNC      581     A

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                 SX5    2                                                    GPO        484    A
GPO       COL5      SX6    B1                                                   GPO        485    A
GPO                 CALL   ESR         EXPAND FP/LVL REFS IN *HB*               GPO        486         I
          -CCGNC
GPO                 CALL   ESR#        EXPAND FP/LVL REFS IN *HB*               CCGNC      582    A
GPO                 SX6    2                                                    GPO        487    A
GPO                 SA6    BSI         BSI = 2                                  GPO        488    A
GPO                 CALL   RTB         GET THE CODED LOOP BODY                  GPO        489         I
          -CCGNC
GPO                 CALL   RTB#        GET THE CODED LOOP BODY                  CCGNC      583    A
GPO                                                                             GPO        490    A
GPO                 SA5    SPC                                                  GPO        491    A
GPO                 SA4    L.TXT                                                GPO        492    A
GPO                 MX6    0                                                    GPO        493    A
GPO                 BX7    X5          PARCEL = SPC                             GPO        494         I
          -CCG7
GPO                 SA7    PARCEL                                               GPO        495         I
          -CCG7
GPO                 SA2    SERF                                                 CCG28        8    A
GPO                 BX7    X2                                                   CCG28        9    A
GPO                 SA7    HO$ER       HO$ER = SERF  */ RESTORE CC VALUE        CCG28       10    A
GPO                 BX7    X5          CG.PC = SPC                              CCG7       101         I
          -CCGNC
GPO                 BX7    X5          CC$PC = SPC                              CCGNC      584    A
GPO                 SA7    CG.PC                                                CCG7       102         I
          -CCGNC
GPO                 SA7    CC$PC                                                CCGNC      585    A
GPO                 SX3    X4-8                                                 GPO        496    A
GPO                 NZ     X3,COL6     IF L.TXT " 8  */ *HB* IS NOT EMPTY       GPO        497    A
GPO                                                                             GPO        498    A
GPO                 SA1    L.SEQ                                                GPO        499    A
GPO                 ALLOC  PIT,X1      ALLOC( PIT , L.SEQ )                     GPO        500    A
GPO                 SX3    X2                                                   GPO        501    A
GPO                 SA2    O.SEQ                                                GPO        502    A
GPO                 MOVE   X1,X2,X3    MOVE (L.SEQ , O.SEQ , O.PIT )            GPO        503    A
GPO                 EQ     COL7                                                 GPO        504    A
GPO                                                                             GPO        505    A
GPO       COL6      CALL   MCG         CODE THE *HB*                            GPO        506         I
          -CCGNC
GPO       COL6      CALL   MCG#        CODE THE *HB*                            CCGNC      586    A
GPO                                                                             GPO        507    A
GPO       *         COMBINE THE CODED INSTRUCTIONS FOR THE *HB* AND LOOP BODY   GPO        508    A
GPO       *         AND CALL *CII* TO CONVERT THEM.                             GPO        509         I
          -CCG1
GPO       *         AND CALL *WII* TO CONVERT THEM.                             CCG1        31    A
GPO                                                                             GPO        510    A
GPO                 SA1    L.SEQ                                                GPO        511    A
GPO                 ALLOC  PIT,X1-1    ALLOC( PIT , L.SEQ-1 )                   GPO        512         I
          -CCG9
GPO                 ALLOC  PIT,X1-2    ALLOC( PIT , L.SEQ-2 )                   CCG9       124    A
GPO                 SB6    B6-B1       OLD(L.PIT) = OLD(L.PIT) - 1              CCG9       125    A
GPO                 SX3    X2+B6                                                GPO        513    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                SA2    O.SEQ                                                GPO          514     A
GPO                SX1    X1+B1                                                CCG18         92     A
GPO                MOVE   X1,X2+B1,X3 MOVE( L.SEQ-1 , O.SEQ+1 , PIT+OLD(L.PIT) )  GPO       515     A
GPO                                                                             GPO          516     A
GPO       COL7     SX6    O.UDB                                                GPO          517     A
GPO                SA6    =XLM        LM = O.UDB                               GPO          518     A
GPO                SA6    O.BLK       O.BLK = O.UDB                            GPO          519     A
GPO                SA6    O.TXT       O.TXT = O.UDB                            GPO          520     A
GPO                SX7    0                                                    GPO          521     A
GPO                SA7    L.BLK       L.BLK = 0                                GPO          522     A
GPO                SA7    L.TXT       L.TXT = 0                                GPO          523     A
GPO                SA7    L.BST       L.BST = 0                                GPO          524     A
GPO                SA7    L.IIT       L.IIT = 0                                GPO          525     A
GPO                SA7    N.GT        N.GT = 0                                 GPO          526     A
GPO                SA1    L.TET                                                GPO          527         I
          -FCC0675
GPO                ALLOC  CFT,X1+B1   ALLOC( CFT , L.TET + 1 )                 GPO          528         I
          -FCC0675
GPO                SETZERO X2,X1      SETZERO( CFT , L.TET+1 )                 GPO          529         I
          -FCC0675
GPO                CALL   CII         CONVERT THE ISSUED INSTRUCTIONS          GPO          530         I
          -CCG1
GPO                CALL   WII         CONVERT THE ISSUED INSTRUCTIONS          CCG1          32         I
          -CCGNC
GPO                CALL   AFT=        DEACTIVATE *BLK*                         GPO          531         I
          -CCGNC
GPO                CALL   WII#        CONVERT THE ISSUED INSTRUCTIONS          CCGNC        587     A
GPO                CALL   AFT#        DEACTIVATE *BLK*                         CCGNC        588     A
GPO                SX7    0                                                    GPO          532     A
GPO                SA7    =XL.CFT     L.CFT = 0                                GPO          533         I
          -FCC0675
GPO                SA7    N.GT        N.GT = 0                                 FCC0675       10     A
GPO                SA7    L.TET       L.TET = 0                                GPO          534     A
GPO                EQ     COL                                                  GPO          535     A
GPO                                                                             GPO          536     A
GPO                QUAL                                                         GPO          537     A
GPO        GPO     TITLE  GPO - CONTROL / INITIALIZATION                       GPO          538     A
GPO       **       GPO - GLOBAL PROGRAM OPTIMIZATION                           GPO          539     A
GPO       *        INITIALIZATION, MAIN LOOP, CODE LINEARIZATION PASS.         GPO          540     A
GPO                                                                             GPO          541     A
GPO        GPO     ENTRY. **                                                   GPO          542         I
          -CCGNC
GPO        CG$GPO  ENTRY. **                                                   CCGNC        589     A
GPO                                                                             GPO          543     A
GPO                QUAL   CODE                                                  GPO          544     A
GPO                                                                             GPO          545     A
GPO                CALL   IMPB        INITIALIZE MASSIO FOR PHASE 2            GPO          546         I
          -CCGNC
GPO                CALL   IMPB#       INITIALIZE MASSIO FOR PHASE 2            CCGNC        590     A
GPO                                                                             GPO          547     A
GPO                SX7    =XF.UDT                                              GPO          548         I
          -CCGNC

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                 SX7    =XF$UDT                                                  CCGNC        591     A
GPO                 CALL   STO                                                      GPO          549     A
GPO                 CALL   AUT          REFORMAT *UDI*                              GPO          550         I
          -CCGNC
GPO                 CALL   AUT#         REFORMAT *UDI*                              CCGNC        592     A
GPO                                                                                 GPO          551     A
GPO                 SX7    =XF.CFA                                                  GPO          552         I
          -CCGNC
GPO                 SX7    =XF$CFA                                                  CCGNC        593     A
GPO                 CALL   STO                                                      GPO          553     A
GPO                 CALL   DGS          DERIVE GRAPH STRUCTURE                      GPO          554         I
          -CCGNC
GPO                 CALL   DGS#         DERIVE GRAPH STRUCTURE                      CCGNC        594     A
GPO                                                                                 GPO          555     A
GPO       *         MOVE GRAPH TABLE DOWN TO BEGIN OF WORKING STORAGE               GPO          556     A
GPO                                                                                 GPO          557     A
GPO                 SA1    L.GST                                                    GPO          558     A
GPO                 SA2    O.GST                                                    GPO          559     A
GPO                 SA3    O.IST                                                    GPO          560     A
GPO                 IX0    X1+X3                                                    GPO          561     A
GPO                 SX6    X3+B1                                                    GPO          562     A
GPO                 SA6    GP           GP = [O.IST] + 1  */ POINT TO GA WORD       GPO          563     A
GPO                 MOVE   X1,X2,X3     MOVE( L.GST , O.GST , O.IST )               GPO          564     A
GPO                 BX7    X0           LM = O.IST + L.GST                          GPO          565     A
GPO                 CALL   STO                                                      GPO          566     A
GPO                                                                                 GPO          567     A
GPO                 CALL   AFT=         ACTIVATE FIRST TABLE ( *BLK* )              GPO          568         I
          -CCGNC
GPO                 CALL   AFT#         ACTIVE *BLK*                                CCGNC        595     A
GPO                                                                                 GPO          569     A
GPO       *         SETUP FIXED TABLES IN LOW MEMORY, LAYOUT -                      GPO          570     A
GPO       *                IST , BIT , UDT , MVL , BVT , SBV , BTT                  GPO          571         I
          -FCC0392
GPO       *                IST , BIT , UDT , MVL , BVT , SBV , BTT , HNT            FCC0392       23         I
          -FCC1319
GPO       *                IST, BIT, LCT, UDT, MVL, BVT, SBV, BTT, HNT              FCC1319       20     A
GPO                                                                                 GPO          572     A
GPO                 SX1    2                                                        FCC0641        9     A
GPO                 RJ     AFB          AFB( 2 )      */ SEARCH TERM FOR GRA/CIS    FCC0641       10     A
GPO                 MX7    -BI.ILP                                                  FCC0641       11     A
GPO                 SA7    X6           [FWA] = MASK(3)  */ SO INIT PARCEL CNT OK   FCC0641       12     A
GPO                 MX7    0                                                        FCC0641       13     A
GPO                 SA7    A7+B1                                                    FCC0641       14     A
GPO                 SA2    CBN                                                      GPO          573         I
          -CCG7
GPO                 SA2    CG.CBN                                                   CCG7         103         I
          -CCGNC
GPO                 SA2    CC$CBN                                                   CCGNC        596     A
GPO                 LX2    1                                                        GPO          574     A
GPO                 SX1    X2+2         L.BIT = 2*CBN + 2                           GPO          575         I
          -FCC1858X
GPO                 SX1    X2+4         L.BIT = 2*CBN + 4                           FCC1858X       4     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                RJ     AFB         AFB( L.BIT )                                GPO          576     A
GPO                SA6    O.BIT       O.BIT = FWA                                 GPO          577     A
GPO                                                                               GPO          578     A
GPO      *         MOVE *BIT* TO LOW CORE AND CLEAR OUT HOLDING BLOCKS WORDS      GPO          579     A
GPO                                                                               GPO          580     A
GPO                SA1    L.BST                                                   GPO          581     A
GPO                SA7    L.BIT                                                   GPO          582     A
GPO                SA2    O.BST                                                   GPO          583     A
GPO                IX0    X7-X1       LHB = L.BIT - L.BST                         GPO          584     A
GPO                IX5    X6+X1                                                   GPO          585     A
GPO                MOVE   X1,X2,X6    MOVE( L.BST , O.BST , O.BIT )               GPO          586     A
GPO                SETZERO X5,X0      FOR I = L.BST TO L.BIT; BIT(I) = 0          GPO          587     A
GPO                SA6    L.BST       L.BST = 0                                   GPO          588     A
GPO                                                                               GPO          589     A
GPO                SA1    L.LCT       L = L.LCT                                   FCC1319       21     A
GPO                SA6    A1          L.LCT = 0     */ CLEAR COUNTER              FCC1319       22     A
GPO                RJ     AFB         AFB( L )                                    FCC1319       23     A
GPO                SA6    O.LCT       O.LCT = FWA                                 FCC1319       24     A
GPO                                                                               FCC1319       25     A
GPO                SA2    N.FTH                                                   CCGA021       31     A
GPO                SX1    X2+B1                                                   CCGA021       32     A
GPO                RJ     AFB         AFB( N.FTH + 1 )                            CCGA021       33     A
GPO                SA6    O.ELT       O.ELT = FWA                                 CCGA021       34     A
GPO      *         MOVE *UDI* TO LOW CORE, ALLOCATE FOR *MVL*.                    GPO          590     A
GPO                                                                               GPO          591     A
GPO       IGO0     SA1    L.UDI                                                   GPO          592     A
GPO                RJ     AFB         AFB( L.UDI )                                GPO          593     A
GPO                SA2    O.UDI                                                   GPO          594     A
GPO                SA6    O.UDT       O.UDT = FWA                                 GPO          595     A
GPO                SA7    L.UDT                                                   GPO          596     A
GPO                MOVE   X1,X2,X6    MOVE( L.UDI , O.UDI , O.UDT )               GPO          597     A
GPO                MX7    0           L.UDI = 0                                   GPO          598     A
GPO                SA7    L.UDI                                                   GPO          599     A
GPO                                                                               GPO          600     A
GPO                SA2    N.VAR                                                   GPO          601     A
GPO                SX1    X2+3                                                    GPO          602     A
GPO                RJ     AFB         ALLOCATE FOR *MVL*                          GPO          603     A
GPO                SA6    O.MVL       O.MVL = FWA                                 GPO          604     A
GPO                                                                               GPO          605     A
GPO      *         ALLOCATE FOR BIT VECTORS, MOVE VECTORS FORMED IN *AUT* TO      GPO          606     A
GPO      *         *BVT*.                                                         GPO          607     A
GPO                                                                               GPO          608     A
GPO                SA1    CBN                                                     GPO          609           I
          -CCG7
GPO                SA1    CG.CBN                                                  CCG7         104           I
          -CCGNC
GPO                SA1    CC$CBN                                                  CCGNC        597     A
GPO                SA2    VL                                                      GPO          610     A
GPO                LX1    2                                                       GPO          611     A
GPO                IX1    X1*X2       L.BVT = CBN * 4*VL  */ BIT VECTORS          GPO          612     A
GPO                RJ     AFB         AFB( L.BVT )                                GPO          613     A
GPO                SA6    O.BVT       O.BVT = FWA                                 GPO          614     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                                                                                GPO          615     A
GPO                 SA1    L.RXI                                                   GPO          616     A
GPO                 SA2    O.RXI                                                   GPO          617     A
GPO                 SX3    X6                                                      GPO          618     A
GPO                 BX7    X1                                                      GPO          619     A
GPO                 SA7    L.BVT       L.BVT = L.RXI */  *BVT* INDEX FOR *FUD*      GPO          620     A
GPO                 MX6    0                                                       GPO          621     A
GPO                 SA6    A1          L.RXI = 0                                   GPO          622     A
GPO                 MOVE   X1,X2,X3    MOVE( L.BVT , O.RXI , O.BVT )               GPO          623     A
GPO                 SA1    =XOPT2                                                  GPO          624         I
          -CCG1
GPO                 ZR     X1,IGO1     IF OPT2 = 0   */ OPT=1                      GPO          625         I
          -CCG1
GPO                 SA1    =XCO.OPT                                                CCG1          33         I
          -CCGNC
GPO                 SA1    =XHO$OPT                                                CCGNC        598     A
GPO                 PL     X1,IGO1     IF OPT " 2                                  CCG1          34     A
GPO                                                                                GPO          626     A
GPO       *         PHASE 2 OF OPT=2 - FORM BIT VECTORS FOR ALL PROGRAM BLOCKS.     GPO          627     A
GPO                                                                                GPO          628     A
GPO                 CALL   FBV         FORM BIT VECTORS                            GPO          629         I
          -CCGNC
GPO                 CALL   FBV#        FORM BIT VECTORS                            CCGNC        599     A
GPO                                                                                GPO          630     A
GPO       *         SETUP ADDRESS"S OF SPECIAL BIT VECTORS                          GPO          631     A
GPO                                                                                GPO          632     A
GPO        IGO1     SA2    VL                                                      GPO          633     A
GPO                 SX0    N.SBV                                                   GPO          634     A
GPO                 IX1    X0*X2                                                   GPO          635     A
GPO                 IX6    X1-X2       L.SBV = VL * ( N.SBV-1 )                    GPO          636     A
GPO                 SA6    L.SBV                                                   GPO          637     A
GPO                 RJ     AFB         AFB( VL * N.SBV )                           GPO          638         I
          -FCC0509
GPO                 IX1    X1+X2                                                   FCC0509        7     A
GPO                 RJ     AFB         AFB( VL * (N.SBV+1) )                       FCC0509        8     A
GPO                 SA1    VL                                                      GPO          639     A
GPO                 SB2    B0          I = 0                                       GPO          640     A
GPO                 SB3    X0                                                      GPO          641     A
GPO        IGO2     SA6    DVA+B2      DVA(I) = FWA                                GPO          642     A
GPO                 SB2    B2+B1       I = I + 1                                   GPO          643     A
GPO                 IX6    X6+X1       FWA = FWA + VL                              GPO          644     A
GPO                 LT     B2,B3,IGO2  IF I < N.SBV                                GPO          645     A
GPO                 SA1    OPT2                                                    GPO          646         I
          -CCG1
GPO                 ZR     X1,COL1     IF OPT2 = 0                                 GPO          647         I
          -CCG1
GPO                 SA1    =XCO.OPT                                                CCG1          35         I
          -CCGNC
GPO                 SA1    =XHO$OPT                                                CCGNC        600     A
GPO                 PL     X1,COL1     IF OPT " 2                                  CCG1          36     A
GPO                                                                                GPO          648     A
GPO       *         ALLOCATE FOR *BTT*                                              GPO          649         I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
          -FCC0392
GPO       *         ALLOCATE FOR *BTT* AND *HNT*                                  FCC0392      24     A
GPO                                                                               GPO         650     A
GPO                 SA1    CBN                                                    GPO         651          I
          -CCG7
GPO                 SA1    CG.CBN                                                 CCG7        105          I
          -CCGNC
GPO                 SA1    CC$CBN                                                 CCGNC       601     A
GPO                 RJ     AFB         AFB( CBN )                                 GPO         652     A
GPO                 SA6    O.BTT       O.BTT = FWA                                GPO         653     A
GPO                 SA5    M.LP                                                   FCC0392      25     A
GPO                 SX1    X5+1                                                   FCC0392      26     A
GPO                 RJ     AFB         AFB( M.LP+1 )                              FCC0392      27     A
GPO                 SA6    O.HNT       O.HNT = FWA                                FCC0392      28     A
GPO        GPO      SPACE  1,14                                                   GPO         654     A
GPO       *         COMPUTE AN UPPER BOUND FOR *BLK* LENGTH                       GPO         655     A
GPO                                                                               GPO         656     A
GPO                 SA1    LM                                                     GPO         657          I
          -RSM2763X
GPO                 SA2    =XO.TEND                                               GPO         658          I
          -RSM2763X
GPO                 IX6    X2-X1       WS = O.TEND - LM                           GPO         659          I
          -RSM2763X
GPO                 AX6    2                                                      GPO         660          I
          -RSM2763X
GPO                 SA6    M.EBL       M.EBL = WS / 4                             GPO         661          I
          -RSM2763X
GPO                 AX5    B1,X6                                                  GPO         662          I
          -RSM2763X
GPO                 IX7    X5+X6       AWS = WS * 3/8                             GPO         663          I
          -RSM2763X
GPO                 SA1    =XM.BBL                                                GPO         664          I
          -RSM2763X
GPO                 SX2    1000B                                                  GPO         665          I
          -RSM2763X
GPO                 IX3    X2-X1                                                  GPO         666          I
          -RSM2763X
GPO                 PL     X3,IGO3                                                GPO         667          I
          -RSM2763X
GPO                 BX2    X1                                                     GPO         668          I
          -RSM2763X
GPO        IGO3     SX2    X2+500B     MIN.BLK = MAX( M.BBL , 1000B ) + 500B      GPO         669          I
          -RSM2763X
GPO                 IX3    X7-X2                                                  GPO         670          I
          -RSM2763X
GPO                 PL     X3,IGO4     IF AWS > MIN.BLK                           GPO         671          I
          -RSM2763X
GPO                                                                               GPO         672          I
          -RSM2763X
GPO                 SA1    =7LGPO                                                 GPO         673          I
          -RSM2763X
GPO                 CALL   PUNT        INSUFFICIENT STORAGE FOR COMPILATION       GPO         674          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

          -RSM2763X
GPO                                                                    GPO          675          I
          -RSM2763X
GPO        IGO4     SX2    14000B                                      GPO          676          I
          -RSM2763X
GPO                 IX6    X7-X2                                       GPO          677          I
          -RSM2763X
GPO                 MI     X6,IGO5     MAX.BLK = MIN( AWS , 14000B )   GPO          678          I
          -RSM2763X
GPO                 BX7    X2                                          GPO          679          I
          -RSM2763X
GPO        IGO5     CALL   IMPC        INITIALIZE MASS I/O FOR PHASE 3 GPO          680          I
          -RSM2763X
GPO                 PRINT  GPOFL,(* CWS= CALLED BY GPO; CWS,MWS =*2Z7),(X6,X7)  RSM2763X   125     A
GPO        IGO3     CALL   CWS=                                        RSM2763X     126          I
          -CCGNC
GPO        IGO3     CALL   CWS#                                        CCGNC        602     A
GPO                 SX1    2000B                                       RSM2763X     127     A
GPO                 SB4    X7          B4 = MWS */ MAX. AVAIL. STORAGE RSM2763X     128     A
GPO                 IX7    X7-X1                                       RSM2763X     129     A
GPO                 AX7    2                                           RSM2763X     130     A
GPO                 SX4    14000B                                      RSM2763X     131     A
GPO                 IX2    X4-X7                                       RSM2763X     132     A
GPO                 SB2    X6          B2 = CWS */ CURRENT WORKING STORAGE RSM2763X 133     A
GPO                 PL     X2,IGO3A    IF (MWS - 2000B)/4 @ 14000B     RSM2763X     134     A
GPO                 BX7    X4                                          RSM2763X     135     A
GPO        IGO3A    SA7    M.EBL       M.EBL = MIN((MWS-2000B)/4,14000B) RSM2763X 136     A
GPO                 SA3    =XM.BBL     */ MAX.BASIC BLOCK SIZE         RSM2763X     137     A
GPO                 PRINT  GPOFL,(* M.BBL =*Z7),(X3)                   RSM2763X     138     A
GPO                 LX3    3                                           RSM2763X     139     A
GPO                 SB3    X3          B3 = 8*M.BBL                    RSM2763X     140     A
GPO                 SX1    10000B                                      RSM2763X     141     A
GPO                 IX2    X3-X1                                       RSM2763X     142     A
GPO                 PL     X2,IGO3B    IF   8*M.BBL > 10000B           RSM2763X     143     A
GPO                 BX3    X1                                          RSM2763X     144     A
GPO                 SB3    X1          RWS = MAX(  8*M.BBL,10000B) */ REASONABLE RSM2763X 145 A
GPO       *                                                STORAGE     RSM2763X     146     A
GPO        IGO3B    GE     B2,B3,IGO5  IF CWS \ RWS                    RSM2763X     147     A
GPO                 GT     B3,B4,IGO4  IF RWS > MWS                    RSM2763X     148     A
GPO                                                                    RSM2763X     149     A
GPO       *                            CWS < RWS @ MWS                 RSM2763X     150     A
GPO                                                                    RSM2763X     151     A
GPO                 IX1    X3-X6                                       RSM2763X     152     A
GPO                 CALL   MOREFL      */ WN = RWS - CWS               RSM2763X     153     A
GPO                 PL     X7,IGO3     IF REQUIRED MEMORY GRANTED      RSM2763X     154     A
GPO                                                                    RSM2763X     155     A
GPO        IGO4     SA1    =7LGPO                                      RSM2763X     156     A
GPO                 CALL   PUNT        GO TO ABORT THE COMPILATION     RSM2763X     157     A
GPO                                                                    RSM2763X     158     A
GPO        IGO5     BX1    X6                                          RSM2763X     159     A
GPO                 PRINT GPOFL,(*CWS,RWS=*2Z7),(X6,X3)                RSM2763X     160     A
GPO                 AX1    3                                           RSM2763X     161     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                AX6    1                                                    RSM2763X    162    A
GPO                IX7    X6-X1       X7 = 3/8 * CWS                           RSM2763X    163    A
GPO                IX2    X4-X7                                                RSM2763X    164    A
GPO                PL     X2,IGO5A                                             RSM2763X    165    A
GPO                BX7    X4                                                   RSM2763X    166    A
GPO       IGO5A    SA7    =XMX.BLN    MX.BLN = MAX BLK.TABLE SIZE              RSM2763X    167    A
GPO                PRINT  GPOFL,(* MAX BLK. LEN =*Z7),(X7)                     RSM2763X    168    A
GPO                CALL   IMPC        MX.BLN  = MIN(3/8*CWS,14000B)            RSM2763X    169         I
          -CCGNC
GPO                CALL   IMPC#       MX.BLN = MIN(3/8*CWS,14000B)             CCGNC       603    A
GPO                                                                            GPO         681    A
GPO       .T       IFNE   TEST,0                                 * TEST MODE * GPO         682    A
GPO                SA1    =XR=FLAG                                             GPO         683         I
          -CCG1
GPO                SA1    =XCO.LO=S                                            CCG1         37         I
          -CCGNC
GPO                SA1    =XHO$LO$S                                            CCGNC       604    A
GPO                ZR     X1,IGO6     IF LIST = 0                              GPO         684    A
GPO                                                                            GPO         685    A
GPO       *        LIST OPTIMIZER MEMORY USEAGE STATISTICS                     GPO         686    A
GPO                                                                            GPO         687    A
GPO                USE    STAT                                                 GPO         688    A
GPO                                                                            GPO         689    A
GPO       IGOA     DATA   1H ,38LOPTIMIZER MEMORY USEAGE STATISTICS            GPO         690    A
GPO       IGOB     DATA   1H ,40LEXEC STMTS  N.BLOCKS   N.LOOPS    N.VARS      GPO         691    A
GPO                DATA   20H   MAX-BLK  N.GRAPHS                              GPO         692    A
GPO                DATA   50L   L.GRAPH L.BIT VEC     L.BLK WRK SPACE L.PRG TXT GPO         693    A
GPO       IGOC     DATA   8L                                                   GPO         694    A
GPO                                                                            GPO         695    A
GPO       ZOD      ROUTINE                                                     GPO         696    A
GPO                RJ     =XCOD                                                GPO         697         I
          -CCG1
GPO                RJ     =XCOD=                                               CCG1         38    A
GPO                LX6    6                                                    GPO         698    A
GPO                SX7    1RB-1R                                               GPO         699    A
GPO                IX6    X6+X7                                                GPO         700    A
GPO                EQ     ZOD                                                  GPO         701    A
GPO                                                                            GPO         702    A
GPO                USE    0                                                    GPO         703    A
GPO       STAT     MACRO  A,B,C,M                                              GPO         704    A
GPO                SA1    A                                                    GPO         705         I
          -CCG1
GPO                SA1    =X_A                                                 CCG1         39    A
GPO                IFC    NE,/B//,2                                            GPO         706    A
GPO                SA2    B                                                    GPO         707    A
GPO                IX1    X1-X2                                                GPO         708    A
GPO                IFC    NE,/M//,1                                            GPO         709    A
GPO                SX1    X1+M                                                 GPO         710    A
GPO                RJ     =X;C                                                 GPO         711    A
GPO                SA6    IGOB+.1                                              GPO         712    A
GPO       .1       SET    .1+1                                                 GPO         713    A
GPO                ENDM                                                        GPO         714    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO       .1       SET    1                                                  GPO          715     A
GPO                LISTL  IGOC,1                                             GPO          716     A
GPO                LISTL  IGOA,IGOB-IGOA                                     GPO          717     A
GPO                LISTL  IGOB,IGOC+1-IGOB                                   GPO          718     A
GPO                STAT   =XN.EXST,,CDD     N.STMTS                          GPO          719         I
          -CCG1
GPO                STAT   NPB,,CDD                                           GPO          720         I
          -CCG1
GPO                STAT   CBN,NPB,CDD,-1    N.LOOPS                          GPO          721         I
          -CCG1
GPO                STAT   N.VAR,,CDD,-1                                      GPO          722         I
          -CCG1
GPO                STAT   N.EXST,,CDD=      N.STMTS                          CCG1          40         I
          -CCGNC
GPO                STAT   N$EXST,,CDD=      N.STMTS                          CCGNC        605     A
GPO                STAT   NPB,,CDD=                                          CCG1          41     A
GPO                STAT   CBN,NPB,CDD=,-1    N.LOOPS                         CCG1          42         I
          -CCG7
GPO                STAT   CG.CBN,NPB,CDD=,-1  N.LOOPS                        CCG7         106         I
          -CCGNC
GPO                STAT   CC$CBN,NPB,CDD=,-1  N.LOOPS                        CCGNC        606     A
GPO                STAT   N.VAR,,CDD=,-1                                     CCG1          43     A
GPO                STAT   M.BBL,,ZOD        MAX BASIC BLK LEN                GPO          723     A
GPO                STAT   N.DG,,COD         N.GRAPHS                         GPO          724         I
          -CCG1
GPO                STAT   N.DG,,COD=        N.GRAPHS                         CCG1          44     A
GPO                STAT   O.BIT,GP,ZOD      L.GRAPH                          GPO          725     A
GPO                STAT   O.BTT,O.BVT,ZOD   L.BIT VEC                        GPO          726     A
GPO                STAT   L.BLK,,ZOD                                         GPO          727     A
GPO                STAT   O.TEND,O.TXT,ZOD  WORK SPACE                       GPO          728         I
          -CCG1
GPO                STAT   O.GLT,O.TXT,ZOD   WORK SPACE                       CCG1          45     A
GPO                STAT   L.PTXT,,ZOD                                        GPO          729     A
GPO                PURGMAC STAT                                              GPO          730     A
GPO                                                                          GPO          731     A
GPO                LISTL  IGOB,IGOC+1-IGOB                                   GPO          732     A
GPO                                                                          GPO          733     A
GPO       IGO6     BSS    0                                                  GPO          734     A
GPO                                                                          GPO          735     A
GPO       .T       ENDIF                                   * TEST MODE *     GPO          736     A
GPO       GPO      SPACE  2                                                  GPO          737     A
GPO      *         SET FLAGS FOR PHASE 3 OF GLOBAL OPTIMIZATION              GPO          738     A
GPO                                                                          GPO          739     A
GPO                SX6    1                                                  GPO          740     A
GPO                SA6    BBV         BBV = 1       */ SET FOR NO BIT VECTORS GPO          741     A
GPO                MX7    UD.MVL                                            GPO          742     A
GPO                LX7    UD.MVL+UD.MVP                                     GPO          743     A
GPO                SA7    BCM         BCM = *MV*    */ SET FOR BLOCK BIT CLEAR GPO         744     A
GPO                                                                          GPO          745     A
GPO      *         SETUP A *BOS* AT THE BEGINNING OF *TXT*                   GPO          746     A
GPO                                                                          GPO          747     A
GPO                ALLOC  TXT,4       ALLOC( TXT , 4 )                       GPO          748     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO              SA5    =XF.RDT+OC.BOS                                    GPO          749     A
GPO              MX6    0            T = O.TXT                            GPO          750     A
GPO              SA6    X2+B1        [T+1] = 0                            GPO          751     A
GPO              BX7    X5                                                GPO          752     A
GPO              SA7    A6+B1        [T+2] = RDT(BOS)                     GPO          753     A
GPO              SA6    A7+B1        [T+3] = 0                            GPO          754     A
GPO              SA1    L.GLT                                             CCG1          46     A
GPO              SX6    X1                                                CCG1          47           I
         -CCG28
GPO              SA6    N.GL0        N.GL0 = L.GLT                        CCG1          48           I
         -CCG28
GPO              SX6    X1-1                                              CCG28         11     A
GPO              SA6    N.GL0        N.GL0 = L.GLT - 1                    CCG28         12     A
GPO       GPO    TITLE  GPO - MAIN LOOP                                  GPO          755     A
GPO      *       PHASE 3 - INNER TO OUTER PASS OVER THE LOOPS TO MOVE OUT GPO          756     A
GPO      *       INVARIANT / STRENGTH REDUCABLE CODE AND ASSIGN REGISTERS. GPO          757     A
GPO                                                                       GPO          758     A
GPO       GPO1   RJ     IGP          INITIALIZE GRAPH PROCESSING          GPO          759     A
GPO              ZR     X6,GPO10     IF N.LP = 0   */ NO LOOPS IN THIS GRAPH GPO       760     A
GPO                                                                       GPO          761     A
GPO       GPO2   RJ     FNL          FIND NEXT LOOP                       GPO          762     A
GPO              RJ     IRP          INITIALIZE REGION PROCESSING         GPO          763     A
GPO                                                                       GPO          764     A
GPO              CLCM   FUD          FORM USE/DEF INFO FOR REGION         GPO          765     A
GPO              SA1    UXR+1                                             GPO          766     A
GPO              ZR     X1,GPO3      IF UXR(2) = 0 */ NO USER EXT REFS    GPO          767     A
GPO              MX7    1                                                 GPO          768     A
GPO              SA7    XRF          XRF = 1S59                           GPO          769     A
GPO              SA2    RCV                                               GPO          770     A
GPO              ZR     X2,GPO3      IF RCV = 0    */ NO REFED COMMON VARIABLES GPO       771     A
GPO              SA4    O.BVT                                             GPO          772     A
GPO              SA5    RDB                                               GPO          773     A
GPO              CALL   EBV          SET REGION DEF BITS FOR COMMON VARS, ETC GPO       774     A
GPO                                                                       GPO          775     A
GPO       GPO3   CLCM   RDD          REMOVE DEAD DEFINITIONS              GPO          776     A
GPO                                                                       GPO          777     A
GPO              RJ     FEI          FORM LIVE ENTRY VECTOR               FCCA513X       8     A
GPO                                                                       FCCA513X       9     A
GPO              CLCM   MII          MARK INVARIANT INSTRUCTIONS          GPO          778     A
GPO                                                                       GPO          779     A
GPO              RJ     IHB          INITIALIZE HOLDING BLOCK             GPO          780     A
GPO              RJ     FXI          FORM EXIT INFORMATION                GPO          781     A
GPO                                                                       GPO          782     A
GPO       GPO3A  CLCM   FII,(STC)    FORM INCREMENT INFO                  GPO          783     A
GPO              SA1    L.IIT                                             GPO          784     A
GPO              BX6    X1           PIC = L.IIT                          GPO          785     A
GPO              SA6    PIC                                               GPO          786     A
GPO              ZR     X1,GPO4      IF PIC = 0    */ NO RECURSIVE DEFS   GPO          787     A
GPO                                                                       GPO          788     A
GPO              CLCM   MIP          MARK INTEGER POLYNOMIALS             GPO          789     A
GPO                                                                       GPO          790     A
GPO       GPO4   SA1    MRA                                               GPO          791     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                SX2    376B                                                GPO          792     A
GPO                SX6    B1+B1       MINC = 2      */ MIN *IP* COST          GPO          793     A
GPO                BX3    X2*X1                                               GPO          794     A
GPO                CX4    X3          NABR = COUNT(MRA&376B)  */ N.AVAIL B-REGS GPO        795     A
GPO                AX4    1                                                   GPO          796     A
GPO                NZ     X4,GPO5     IF NABR \ 2                             GPO          797     A
GPO                SX6    3           MINC = 3                                GPO          798     A
GPO        GPO5    SA6    MINC                                                GPO          799     A
GPO                CLCM   MIE,(IIC)   MOVE INV/SR INSTRUCTIONS TO *HB*        GPO          800     A
GPO                MX6    0                                                   GPO          801     A
GPO                CALL   SHB         SQUEEZE HOLDING BLOCK                   GPO          802     A
GPO                SA2    NTP                                                 GPO          803     A
GPO                ZR     X2,GPO6     IF NTP = 0    */ NO TERMINAL *IP*"S     GPO          804     A
GPO                                                                           GPO          805     A
GPO                RJ     CIF         COLLECT INCREMENT FORMULAS              GPO          806     A
GPO                SA2    NTP                                                 FCC1710X       6     A
GPO                ZR     X2,GPO6     IF NTP = 0                              FCC1710X       7     A
GPO                                                                           GPO          807     A
GPO                CLCM   FIM,(STC)   FORM *TP* INCREMENT MODS                GPO          808     A
GPO                SA1    L.MOD                                               GPO          809     A
GPO                SX2    X1-4                                                GPO          810     A
GPO                ZR     X2,GPO6     IF L.MOD = 4  */ NO VARIABLE INCR CODE  GPO          811     A
GPO                SX6    B1                                                  GPO          812     A
GPO                CALL   SHB         SQUEEZE VARIABLE INCREMENT CODE         GPO          813     A
GPO                                                                           GPO          814     A
GPO        GPO6    SX6    4           L.TXT = 4                               GPO          815     A
GPO                SA6    L.TXT                                               GPO          816     A
GPO                CLCM   UPB,(NIS,BM,NTET)   UPDATE MODIFIED BLOCKS          GPO          817     A
GPO                RJ     ATT         ADJUST *TET*                            FCC0514       25     A
GPO                                                                           GPO          818     A
GPO        GPO7    RJ     CBB         COMBINE BASIC BLOCKS                    GPO          819         I
           -FCC0646
GPO                RJ     CBB         COMBINE BASIC BLOCKS                    FCC0646       12     A
GPO                                                                           GPO          820     A
GPO                CALL   GRA         ASSIGN REGISTERS, CODE THE LOOP         GPO          821         I
           -CCGNC
GPO                CALL   GRA#        ASSIGN REGISTERS, CODE THE LOOP         CCGNC        607     A
GPO                                                                           GPO          822     A
GPO                RJ     TRP         TERMINATE REGION PROCESSING             GPO          823         I
           -FCC0646
GPO        GPO7    RJ     TRP         TERMINATE REGION PROCESSING             FCC0646       13     A
GPO                SA2    N.LP                                                GPO          824     A
GPO                SX7    X2-1        N.LP = N.LP - 1                         GPO          825     A
GPO                SA7    A2+                                                 GPO          826     A
GPO                NZ     X7,GPO2     IF N.LP " 0                             GPO          827     A
GPO                                                                           GPO          828     A
GPO                SX7    4           EMF = 4       */ EXPAND STT"S IN OUTER LPS GPO       829     A
GPO                SA7    EMF                                                 GPO          830     A
GPO                RJ     GNG         GET NEXT GRAPH                          GPO          831     A
GPO                NZ     X6,GPO1     IF NOT END OF GRAPHS                    GPO          832         I
           -FCC0514
GPO                EQ     *+1S17            GRAPH TABLE ERROR                 GPO          833         I

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

          -FCC0514
GPO                EQ     GPO1                                                    FCC0514      26     A
GPO        GPO     SPACE  2,14                                                    GPO         834     A
GPO       *        PROCESS BLOCKS OUTSIDE OF ALL LOOPS AND CONVERT THEM TO CODE.  GPO         835     A
GPO                                                                               GPO         836     A
GPO        GPO10   MX6    -UD.STCP                                                GPO         837     A
GPO                SX7    B0                                                      FCC0514      27     A
GPO                SA6    PSLC        PSLC = 1      */ INDICATE OUT OF ALL LOOPS  FCC0514      28           I
          -FCC0675
GPO                SA7    HBI         HBI = 0       */ OUTSIDE OF ALL LOOPS       FCC0675      11     A
GPO                SA7    NBIP        NBIP = 0                                    FCC0514      29     A
GPO                SA6    BCM         BCM = *REGION CLEAR MASK*                   GPO         838     A
GPO                                                                               GPO         839     A
GPO        GPO11   SA1    IBA                                                     GPO         840     A
GPO                SA2    X1          IP = [IBA]                                  GPO         841     A
GPO                SX6    X1+B1       IBA = IBA + 1                               GPO         842     A
GPO                ZR     X2,GPO12    IF [IP] = 0   */ END OF INTERVAL LIST       GPO         843     A
GPO                LX2    -IP.NODEP                                               GPO         844     A
GPO                SA6    A1                                                      GPO         845     A
GPO                SX7    X2          NN = NODE[IP]                               GPO         846     A
GPO                SA7    A1+B1                                                   GPO         847     A
GPO                CALL   CIC         CONVERT INTERVAL TO CODE                    GPO         848     A
GPO                EQ     GPO11                                                   GPO         849     A
GPO                                                                               GPO         850     A
GPO       *        ADJUST *GL* TABLE FOR GL"S ADDED FROM MULTIPLE PRED IF LOOPS.  GPO         851     A
GPO                                                                               GPO         852     A
GPO        GPO12   SA1    N.GL0                                                   GPO         853     A
GPO                SA2    N.GL                                                    GPO         854           I
          -CCG1
GPO                IX1    X1-X2       L = N.GL0 - N.GL                            GPO         855           I
          -CCG1
GPO                SA2    L.GLT                                                   CCG1         49     A
GPO                IX1    X1-X2       L = N.GL0 - L.GLT                           CCG1         50     A
GPO                ZR     X1,GPO13    IF L = 0      */ NO GL"S ADDED              GPO         856           I
          -CCG28
GPO                ALLOC  TEND-1,X1   ALLOC( TEND-1 , L )                         GPO         857           I
          -CCG1     -CCG28
GPO                SA5    =XO.GLT                                                 GPO         858           I
          -CCG1     -CCG28
GPO                SX7    B6                                                      GPO         859           I
          -CCG1     -CCG28
GPO                SA7    A3                                                      GPO         860           I
          -CCG1     -CCG28
GPO                IX6    X5-X1       O.GLT = O.GLT - L                           GPO         861           I
          -CCG1     -CCG28
GPO                SA6    A5                                                      GPO         862           I
          -CCG1     -CCG28
GPO                SX7    X6-1        LW = O.GLT - 1                              GPO         863           I
          -CCG1     -CCG28
GPO                SA7    =XLWAWORK   LWAWORK = LW                                GPO         864           I
          -CCG1     -CCG28
GPO                SA7    =XO.TEND    O.TEND = LW                                 GPO         865           I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

          -CCG1     -CCG28
GPO                 ALLOC  GLT,X1      ALLOC( GLT , L )                          CCG1          51          I
          -CCG28
GPO                 MI     X1,GPO13    IF L < 0      */ NO GL"S ADDED            CCG28         13     A
GPO                 ALLOC  GLT,X1+B1   ALLOC( GLT , L+1 )                        CCG28         14     A
GPO        GPO      SPACE  2,20                                                  GPO          866     A
GPO       *         PHASE 4 - LINEARIZE THE CODE BLOCKS TO BRING THEM BACK TO    GPO          867     A
GPO       *         SOURCE PROGRAM ORDER AND OUTPUT THEM.                        GPO          868     A
GPO                                                                              GPO          869     A
GPO                 DESCRIBE BI.,60                                              GPO          870     A
GPO        BEF      DEQU   BVI         BN OF BLOCK THAT COMES BEFORE CURRENT BLOCK GPO        871     A
GPO                                                                              GPO          872     A
GPO        GPO13    SA1    CBN                                                   GPO          873          I
          -CCG7
GPO        GPO13    SA1    CG.CBN                                                CCG7         107          I
          -CCGNC
GPO        GPO13    SA1    CC$CBN                                                CCGNC        608     A
GPO                 SA2    O.BIT                                                 GPO          874     A
GPO                 MX0    -BI.BEFL                                              GPO          875     A
GPO                 SB2    B1+B1                                                 GPO          876     A
GPO                 LX0    BI.BEFP                                               GPO          877     A
GPO                 SB3    B2          I = 2                                     GPO          878          I
          -FCC0408
GPO                 SB3    B0          I = 0                                     FCC0408       10     A
GPO                 LX1    1                                                     GPO          879     A
GPO                 SB4    X1          L = 2*CBN                                 GPO          880     A
GPO                 SA3    NPB                                                   GPO          881     A
GPO                 SB7    X2                       (B7) = [O.BIT]               GPO          882     A
GPO                                                                              GPO          883     A
GPO        GPO14    SA5    B7+B3       BEF[BIT(I)] = 0                           GPO          884     A
GPO                 SB3    B3+B2       I = I + 2                                 GPO          885     A
GPO                 BX6    X0*X5                                                 GPO          886     A
GPO                 SA6    A5                                                    GPO          887     A
GPO                 LT     B3,B4,GPO14 IF I < L                                  GPO          888     A
GPO                                                                              GPO          889     A
GPO       *         SEARCH FOR HOLDING BLOCKS THAT WERE CODED ( NOT COALESCED )  GPO          890     A
GPO       *         AND SET THE *BEF* FIELD FOR THE HEADER BLOCK THAT COMES      GPO          891     A
GPO       *         AFTER THE *HB* IN THE CODE OUTPUT.                           GPO          892     A
GPO                                                                              GPO          893     A
GPO                 LX3    1                                                     GPO          894     A
GPO                 SB3    X3          I = 2*NPB                                 GPO          895     A
GPO                                                                              GPO          896     A
GPO        GPO15    SB3    B3+B2       I = I + 2                                 GPO          897     A
GPO                 SA5    B7+B3       BIW = BIT(I)                              GPO          898     A
GPO                 GE     B3,B4,GPO18 IF I \ L                                  GPO          899     A
GPO                 LX5    59-BI.CBP                                             GPO          900     A
GPO                 PL     X5,GPO15    IF ^CB[BIW]                               GPO          901     A
GPO                                                                              GPO          902     A
GPO                 LX5    1+BI.CBP-BI.PBNP                                      GPO          903     A
GPO                 SX7    B3          P = I                                     GPO          904     A
GPO                 SB6    X5          J = PBN[BIW]  */ BN OF HEADER NODE        GPO          905          I
          -FCC0448

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                LX5    59-BI.FJP+BI.PBNP                                    FCC0408      11          I
          -FCC0448
GPO                PL     X5,GPO16    IF ^FJ[BIW]   */ ^ FALL THROUGH TO HEADER FCC0408      12          I
          -FCC0448
GPO                SB6    0           J = 0         */ ADD *HB* TO EXIT NODE LIST FCC0408      13          I
          -FCC0448
GPO                                                                             GPO         906          I
          -FCC0448
GPO                SB6    X5          J = PBN[BIW]  */ BN OF HEADER NODE        FCC0448      32     A
GPO                SA4    B7+B6       BJW = BIT(J)                              FCC0448      33     A
GPO                LX4    59-BI.RBP                                             FCC0448      34     A
GPO                MI     X4,GPO15A   IF RB[BJW]    */ HEADER IS A PROGRAM BLOCK FCC0448      35     A
GPO                LX4    BI.RBP-BI.FJP                                         FCC0448      36     A
GPO                MI     X4,GPO17    IF FJ[BJW]    */ *HB* IS REMOTE           FCC0448      37     A
GPO                                                                             FCC0448      38     A
GPO       GPO15A   LX5    59-BI.FJP+BI.PBNP                                     FCC0448      39     A
GPO                PL     X5,GPO16    IF ^FJ[BIW]   */ ^ FALL THROUGH TO HEADER FCC0448      40     A
GPO                SB6    0           J = 0         */ ADD *HB* TO EXIT NODE LIST FCC0448      41          I
          -FCC0675
GPO                SB6    B0          J = 0         */ ADD *HB* TO EXIT NODE LIST FCC0675      12     A
GPO       GPO16    SA4    B7+B6       BJW = BIT(J);  JL = J                     GPO         907     A
GPO                LX4    -BI.BEFP                                              GPO         908     A
GPO                SB6    X4          J = BEF[BJW]                              GPO         909     A
GPO                NZ     B6,GPO16    IF J " 0                                  GPO         910     A
GPO                                                                             GPO         911     A
GPO                BX6    X4+X7       BEF[BIT(JL)] = P  */ LINK HEADER TO *HB*  GPO         912     A
GPO                LX6    BI.BEFP                                               GPO         913     A
GPO                SA6    A4                                                    GPO         914     A
GPO                EQ     GPO15                                                 GPO         915     A
GPO                                                                             FCC0448      42     A
GPO       *        *HB* FROM NEST OF *FTH* LOOPS, MAKE *HB* COME BEFORE HEADER. FCC0448      43     A
GPO                                                                             FCC0448      44     A
GPO       GPO17    LX5    BI.PBNP                                               FCC0448      45     A
GPO                SX6    B6                                                    FCC0448      46     A
GPO                LX6    BI.BEFP                                               FCC0448      47     A
GPO                BX6    X6+X5       BEF[BIT(I)] = J  */ HB BEFORE HEADER      FCC0448      48     A
GPO                SB5    B0          K = 0                                     FCC0448      49     A
GPO                                                                             FCC0448      50     A
GPO       GPO17A   SA4    B7+B5       BKW = BIT(K);  KL = K                     FCC0448      51     A
GPO                LX4    -BI.BEFP                                              FCC0448      52     A
GPO                SB5    X4          K = BEF[BKW]                              FCC0448      53     A
GPO                NE     B5,B6,GPO17A      IF K " J                            FCC0448      54     A
GPO                                                                             FCC0448      55     A
GPO                SA6    A5                                                    FCC0448      56     A
GPO                LX4    BI.BEFP                                               FCC0448      57     A
GPO                BX4    X0*X4                                                 FCC0448      58     A
GPO                BX7    X4+X7       BEF[BIT(KL)] = I  */ LINK PREV TO HB      FCC0448      59     A
GPO                SA7    A4                                                    FCC0448      60     A
GPO                EQ     GPO15                                                 FCC0448      61     A
GPO                                                                             GPO         916     A
GPO       *        FINAL SCAN OF PROGRAM BLOCKS IN *BIT* TO FORM A TABLE OF    GPO         917     A
GPO       *        RANDOM INDEX WORDS OF THE BLOCKS IN THEIR OUTPUT ORDER.     GPO         918          I

          0        1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

          -CCG0013
GPO       *         ADDRESSES OF THE BLOCKS IN THEIR OUTPUT ORDER.                    CCG0013        6     A
GPO                                                                                 GPO          919     A
GPO        GPO18    SA4    O.UDT                                                    GPO          920     A
GPO                 SB3    B0          I = 0                                         GPO          921     A
GPO                 SA0    X4          SI = [O.UDT]  */ RIW STORE INDEX              GPO          922     A
GPO                 SB4    X3+B2       L = 2*NPB + 2                                 GPO          923     A
GPO                 EQ     GPO20                                                     GPO          924     A
GPO                                                                                 GPO          925     A
GPO        GPO19    SA4    A5+B1                                                     GPO          926           I
          -CCG0013
GPO                 BX6    X4          [SI] = BIT(I+1)                               GPO          927           I
          -CCG0013
GPO        GPO19    SX6    A5+B1                                                     CCG0013        7     A
GPO                 SA6    A0                                                        GPO          928     A
GPO                 SA0    A0+B1       SI = SI + 1                                   GPO          929     A
GPO                                                                                 GPO          930     A
GPO        GPO20    SB3    B3+B2       I = I + 2                                     GPO          931     A
GPO                 SA5    B7+B3       BIW = BIT(I)                                  GPO          932     A
GPO                 GE     B3,B4,GPO23 IF I \ L                                      GPO          933     A
GPO                 LX5    -BI.BEFP                                                  GPO          934     A
GPO                 SX2    X5          JF = BEF[BIW]                                 GPO          935     A
GPO                 LX5    59-BI.CBP+BI.BEFP                                         GPO          936     A
GPO                 PL     X5,GPO20    IF ^CB[BIW]   */ NOT CODED                    GPO          937     A
GPO                 ZR     X2,GPO19    IF JF = 0     */ NO HEADER NODE BEFORE        GPO          938     A
GPO                                                                                 GPO          939     A
GPO       *         SEARCH TO END OF *BEF* CHAIN, COUNT NUMBER OF BLOCKS BEFORE.      GPO          940     A
GPO                                                                                 GPO          941     A
GPO                 SB6    X2          J = JF                                        GPO          942     A
GPO                 SB5    B1          K = 1                                         GPO          943     A
GPO        GPO21    SA4    B7+B6       BJW = BIT(J)                                  GPO          944     A
GPO                 SB5    B5+1        K = K + 1                                     GPO          945     A
GPO                 LX4    -BI.BEFP                                                  GPO          946     A
GPO                 SB6    X4          J = BEF[BJW]                                  GPO          947     A
GPO                 NZ     B6,GPO21    IF J " 0                                      GPO          948     A
GPO                                                                                 GPO          949     A
GPO       *         RESCAN THE LIST AND STORE *RI* WORDS IN REVERSE ORDER.            GPO          950     A
GPO                                                                                 GPO          951     A
GPO                 SA3    A5+B1                                                     GPO          952           I
          -CCG0013
GPO                 SA0    A0+B5       SI = SI + K                                   GPO          953     A
GPO                 SB6    X2+B1       J = JF                                        GPO          954     A
GPO                 BX7    X3                                                        GPO          955           I
          -CCG0013
GPO                 SX7    A5+B1                                                     CCG0013        8     A
GPO                 SA7    A0-B1       SJ = SI - 1;  [SJ] = BIT(I+1)                 GPO          956     A
GPO        GPO22    SA3    B7+B6                                                     GPO          957           I
          -CCG0013
GPO                 SA4    A3-B1                                                     GPO          958           I
          -CCG0013
GPO                 BX7    X3                                                        GPO          959           I
          -CCG0013

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                                                                                  CCG0013        9     A
GPO       GPO22    SX7    B7+B6                                                      CCG0013       10     A
GPO                SA4    X7-1                                                       CCG0013       11     A
GPO                SA7    A7-B1       SJ = SJ - 1;  [SJ] = BIT(J+1)                  GPO          960     A
GPO                LX4    -BI.BEFP                                                   GPO          961     A
GPO                SB6    X4+B1       J = BEF[BIT(J)]                                GPO          962     A
GPO                NE     B6,B1,GPO22 IF J " 0                                       GPO          963     A
GPO                                                                                  GPO          964     A
GPO                EQ     GPO20                                                      GPO          965     A
GPO                                                                                  GPO          966     A
GPO       GPO23    MX7    0                                                          GPO          967         I
          -FCC0408
GPO                SA7    A0          [SI] = 0                                       GPO          968         I
          -FCC0408
GPO       *        ADD HOLDING BLOCKS FOR FALL THROUGH TO HEADER LOOPS TO LIST        FCC0408       14     A
GPO                                                                                  FCC0408       15     A
GPO       GPO23    SB6    0           J = 0                                          FCC0408       16     A
GPO                                                                                  FCC0408       17     A
GPO       GPO24    SA5    B7+B6       BIW = BIT(J)                                   FCC0408       18     A
GPO                LX5    -BI.BEFP                                                   FCC0408       19     A
GPO                SB6    X5          J = BEF[BIW]                                   FCC0408       20     A
GPO                ZR     B6,GPO24A   IF J = 0      */ END OF CHAIN                  FCC0408       21         I
          -FCC0675
GPO                ZR     B6,GPO25    IF J = 0      */ END OF CHAIN                  FCC0675       13     A
GPO                SB5    B6+B1                                                      FCC0408       22     A
GPO                SA4    B7+B5                                                      FCC0408       23         I
          -CCG0013
GPO                BX7    X4          [SI] = BIT(J+1)                                FCC0408       24         I
          -CCG0013
GPO                SX7    B7+B5                                                      CCG0013       12     A
GPO                SA7    A0                                                         FCC0408       25     A
GPO                SA0    A0+B1       SI = SI + 1                                    FCC0408       26     A
GPO                EQ     GPO24                                                      FCC0408       27     A
GPO       GPO      SPACE  1,10                                                       FCC0408       28     A
GPO       GPO24A   MX7    0                                                          FCC0408       29         I
          -FCC0675
GPO       *        ALLOCATE TABLES FOR *CII* AND ASSIGN *CA"S* TO THE IT.S IN TET     FCC0675       14         I
          -CCG1
GPO       *        ALLOCATE TABLES FOR *WII* AND ASSIGN *CA"S* TO THE IT"S IN TET     CCG1          52     A
GPO                                                                                  FCC0675       15     A
GPO       GPO25    MX7    0                                                          FCC0675       16     A
GPO                SA7    A0          [SI] = 0      */ TERMINATE THE *RIW* LIST      FCC0408       30     A
GPO                SA7    =XPARCEL    PARCEL = 0                                     GPO          969         I
          -CCG7
GPO                SA7    =XCG.PC     CG.PC = 0                                      CCG7         108         I
          -CCGNC
GPO                SA7    =XCC$PC     CC$PC = 0                                      CCGNC        609     A
GPO                SA2    M.PL                                                       GPO          970     A
GPO                ALLOC  PIT,X2+1    ALLOC( PIT , M.PL+1 )                          GPO          971     A
GPO                LX1    2                                                          GPO          972     A
GPO                ALLOC  TXT,X1      ALLOC( TXT , 4*(M.PL+1) )                      GPO          973     A
GPO                SA1    L.TET                                                      GPO          974     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                ALLOC  CFT,X1+B1   ALLOC( CFT , L.TET+1 )                          GPO          975          I
          -FCC0675
GPO                SETZERO  X2,X1     SETZERO( CFT , L.TET+1 )                        GPO          976          I
          -FCC0675
GPO                ZR     X1,GPO35    IF L.TET = 0  */ NO IT. S                       FCC0675       17     A
GPO       MAXT     DEQU   BVI         MAX IT. S IN LOOP                               FCC0675       18     A
GPO                                                                                   FCC0675       19     A
GPO                SA2    O.BIT                                                       FCC0675       20     A
GPO                SA3    L.BIT                                                       FCC0675       21     A
GPO                SA4    NPB                                                         FCC0675       22     A
GPO                MX7    0                                                           FCC0675       23     A
GPO                SA7    X2          BIT(0) = 0                                      FCC0675       24     A
GPO                MX0    -BI.MAXTL                                                   FCC0675       25     A
GPO                LX4    1                                                           FCC0675       26     A
GPO                SB3    X4          I = 2*NPB                                       FCC0675       27     A
GPO                SB4    X3                                                          FCC0675       28     A
GPO       GPO26    SA5    X2+B3                                                       FCC0675       29     A
GPO                BX6    X0*X5       MAXT[BIT(I)] = 0                                FCC0675       30     A
GPO                SB3    B3+2        I = I + 2                                       FCC0675       31     A
GPO                SA6    A5                                                          FCC0675       32     A
GPO                LT     B3,B4,GPO26 IF I < L.BIT                                    FCC0675       33     A
GPO                                                                                   FCC0675       34     A
GPO       *        FORM SORT TABLE AND GROUP IT.S BY THE LOOP THEY ARE DEFINED IN     FCC0675       35     A
GPO                                                                                   FCC0675       36     A
GPO                ALLOC  CFT,X1+B1   ALLOC( CFT , L.TET+1 )                          FCC0675       37     A
GPO                SA5    O.TET                                                       FCC0675       38     A
GPO                SB3    X1-1        LIM = L.TET                                     FCC0675       39     A
GPO                MX0    -T.HBNL                                                     FCC0675       40     A
GPO                MX7    0                                                           FCC0675       41     A
GPO                LX0    T.HBNP                                                      FCC0675       42     A
GPO                SA7    X2          CFT(0) = 0;  J = 1                              FCC0675       43     A
GPO                SB2    B0          I = 0                                           FCC0675       44     A
GPO                SA5    X5                                                          FCC0675       45     A
GPO                                                                                   FCC0675       46     A
GPO       GPO27    BX6    -X0*X5                                                      FCC0675       47     A
GPO                LX5    59-T.EQVP                                                   FCC0675       48     A
GPO                ZR     X6,GPO28    IF HBN[TET(I)] = 0  */ NOT MATERIALIZED         FCC0675       49     A
GPO                MI     X5,GPO28    IF EQV[TET(I)]                                  FCC0675       50     A
GPO                SX1    B2+                                                         FCC0675       51     A
GPO                BX7    X6+X1                                                       FCC0675       52     A
GPO                SA7    A7+B1       J = J + 1;  CFT(J) = HBN[TET(I)] ! I            FCC0675       53     A
GPO       GPO28    SB2    B2+B1       I = I + 1                                       FCC0675       54     A
GPO                SA5    A5+B1                                                       FCC0675       55     A
GPO                LT     B2,B3,GPO27 IF I < LIM                                      FCC0675       56     A
GPO                                                                                   FCC0675       57     A
GPO                SB7    X2+B1       FWA = CFT(1)                                    FCC0675       58     A
GPO                SB6    A7+B1                                                       FCC0675       59     A
GPO                SX1    B6-B7       LEN = CFT(I+1) - FWA                            FCC0675       60     A
GPO                SA0    A7                        (A0) = CFT(N)                     FCC0675       61     A
GPO                ZR     X1,GPO33    IF LEN = 0                                      FCC0675       62     A
GPO                SX6    X1+B1                                                       FCC0675       63     A
GPO                SA6    A3          L.CFT = LEN + 1                                 FCC0675       64     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                CALL   SHL         SORT( CFT , LEN )                          FCC0675      65        I
          -CCG1
GPO                CALL   SHL=        SORT( CFT , LEN )                          CCG1         53        I
          -CCGNC
GPO                CALL   SHL#        SORT( CFT , LEN )                          CCGNC       610    A
GPO                                                                               FCC0675      66    A
GPO       *        SCAN THE SORTED TABLE BACKWARDS ( OUTERMOST LOOPS FIRST ) AND  FCC0675      67    A
GPO       *        ASSIGN *CA"S* TO THE IT. S, AND ACCUMULATE THE AMOUNT OF       FCC0675      68    A
GPO       *        STORAGE USED.                                                  FCC0675      69    A
GPO                                                                               FCC0675      70    A
GPO                SA5    A0          I = N;  CI = CFT(N)                        FCC0675      71    A
GPO                MX0    -T.CAL                                                 FCC0675      72    A
GPO                SA3    O.BIT                                                  FCC0675      73    A
GPO                SA4    O.TET                                                  FCC0675      74    A
GPO                SX7    B0          C = 0                                      FCC0675      75    A
GPO                SX1    B1                                                     FCC0675      76    A
GPO                SB5    B0          MIT = 0       */ MAX IT. S                 FCC0675      77    A
GPO                LX1    T.FAP                                                  FCC0675      78    A
GPO                SB6    X3                        (B6) = O.BIT                 FCC0675      79    A
GPO                SB7    X4                        (B7) = O.TET                 FCC0675      80    A
GPO                SB4    T.HBNP                                                 FCC0675      81    A
GPO                AX2    B4,X5       LHB = HBN[CI]                              FCC0675      82    A
GPO                                                                               FCC0675      83    A
GPO        GPO30   AX3    B4,X5       H = HBN[CI]                                FCC0675      84    A
GPO                BX6    X2-X3                                                  FCC0675      85    A
GPO                ZR     X6,GPO32    IF H = LHB    */ SAME LOOP                 FCC0675      86    A
GPO                                                                               FCC0675      87    A
GPO                SA4    B6+X2                                                  FCC0675      88    A
GPO                BX6    X4+X7       MAXT[BIT(LHB)] = C  */ SAVE OLD MAX        FCC0675      89    A
GPO                SA6    A4+                                                    FCC0675      90    A
GPO       +        SB2    X7                                                     FCC0675      91    A
GPO                LE     B2,B5,*+1   MIT = MAX( MIT , C )                       FCC0675      92    A
GPO                SB5    B2                                                     FCC0675      93    A
GPO                BX3    -X3                                                    FCC0675      94    A
GPO                SB2    X3                                                     FCC0675      95    A
GPO                                                                               FCC0675      96    A
GPO        GPO31   SX2    X2-2        LHB = LHB - 2                              FCC0675      97    A
GPO                SA4    B6+X2                                                  FCC0675      98    A
GPO                MX3    -BI.OLNL                                               FCC0675      99    A
GPO                LX4    -BI.OLNP                                               FCC0675     100    A
GPO                BX3    -X3*X4      J = OLN[BIT(LHB)]                          FCC0675     101    A
GPO                SA3    B6+X3                                                  FCC0675     102    A
GPO                LX3    -BI.MAXTP                                              FCC0675     103    A
GPO                SX7    X3          C = MAXT[BIT(J)]  */ SET BASE *CA*         FCC0675     104    A
GPO                SX6    X2+B2                                                  FCC0675     105    A
GPO                ZR     X6,GPO32    IF LHB = H                                 FCC0675     106    A
GPO                LX4    BI.OLNP                                                FCC0675     107    A
GPO                BX6    X4+X7       MAXT[BIT(OLN)] = C  */ PROPAGATE TO INER LP FCC0675     108    A
GPO                SA6    A4                                                     FCC0675     109    A
GPO                EQ     GPO31                                                  FCC0675     110    A
GPO                                                                               FCC0675     111    A
GPO       *        ASSIGN A *CA* TO THIS *TET* ENTRY                              FCC0675     112    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                                                                                  FCC0675     113     A
GPO       GPO32    SA4    B7+X5       I = SETX([CI])                                 FCC0675     114     A
GPO                SA5    A5-B1       CI = CI - 1                                    FCC0675     115     A
GPO                BX6    X1+X7                                                      FCC0675     116     A
GPO                BX3    X0*X4                                                      FCC0675     117     A
GPO                BX6    X6+X3       (FA,CA)TET(I) = (1,C)                          FCC0675     118     A
GPO                SA6    A4                                                         FCC0675     119     A
GPO                SX7    X7+1        C = C + 1                                      FCC0675     120     A
GPO                NZ     X5,GPO30    IF [CI] " 0   */ NOT END OF SORT TABLE         FCC0675     121     A
GPO                                                                                  FCC0675     122     A
GPO       +        SB2    X7                                                         FCC0675     123     A
GPO                GE     B2,B5,*+1   N.IT = MAX( MIT , C )                          FCC0675     124     A
GPO                SX7    B5                                                         FCC0675     125     A
GPO                SA7    =XN.IT                                                     FCC0675     126         I
          -CCGNC
GPO                SA7    =XN$IT                                                     CCGNC       611     A
GPO                                                                                  FCC0675     127     A
GPO       GPO33    SX7    0                                                          FCC0675     128     A
GPO                SA7    =XL.CFT     L.CFT = 0                                      FCC0675     129     A
GPO       GPO      SPACE  1,14                                                       GPO         977     A
GPO       *        LOOP THROUGH THE TABLE OF *RI* WORDS AND CALL *CII* TO            GPO         978         I
          -CCG1
GPO       *        LOOP THROUGH THE TABLE OF *RI* WORDS AND CALL *WII* TO            CCG1         54     A
GPO       *        CONVERT THE BLOCKS TO COMPASS INSTRCUTIONS, ASSIGN ADDRESS"S      GPO         979         I
          -FCC0675
GPO       *        AND PACK THE TEMPORARIES.                                         GPO         980         I
          -FCC0675
GPO                                                                                  GPO         981         I
          -FCC0675
GPO       GPO25    SA1    O.UDT                                                      GPO         982         I
          -FCC0675
GPO       *        CONVERT THE BLOCKS TO COMPASS INSTRUCTIONS AND ASSIGN             FCC0675     130     A
GPO       *        ADDRESS"S.                                                        FCC0675     131     A
GPO                                                                                  FCC0675     132     A
GPO       GPO35    SA1    O.UDT                                                      FCC0675     133     A
GPO                SA3    O.PIT                                                      GPO         983     A
GPO                SA5    X1          RIW = [SI]                                     GPO         984         I
          -CCG0013
GPO                ZR     X5,GPO26    IF RIW = 0    */ END OF TABLE                  GPO         985         I
          -FCC0675 -CCG0013
GPO                ZR     X5,GPO36    IF RIW = 0    */ END OF TABLE                  FCC0675     134         I
          -CCG0013
GPO                SA5    X1          RIWA = [SI]                                    CCG0013      13     A
GPO                ZR     X5,GPO36    IF RIWA = 0     */ END OF TABLE                CCG0013      14     A
GPO                SA5    X5          RIW = [RIWA]                                   CCG0013      15     A
GPO                LX5    -RI.LENP                                                   GPO         986     A
GPO                SX6    X1+B1       SI = SI + 1                                    GPO         987     A
GPO                SA6    A1                                                         GPO         988     A
GPO                SX7    X5          L.PIT = LEN[RIW]                               GPO         989     A
GPO                LX5    RI.LENP                                                    GPO         990     A
GPO                SA7    L.PIT                                                      GPO         991     A
GPO                CALL   GMB         READ BLOCK TO *PIT*                            GPO         992         I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

          -CCGNC
GPO                CALL   GMB#        READ BLOCK INTO *PIT*                          CCGNC        612     A
GPO                SA1    O.PIT       R1 = [O.PIT]                                   GPO          993     A
GPO                SA2    X1                                                         GPO          994     A
GPO                SX6    X2          BN = IH[R1]   */ SET *BN* FOR TET PACKING      GPO          995     A
GPO                SA6    BN                                                         GPO          996     A
GPO                CALL   CII         CONVERT THE INSTRUCTIONS                       GPO          997         I
          -CCG1
GPO                CALL   WII         CONVERT THE INSTRUCTIONS                       CCG1          55         I
          -CCGNC
GPO                CALL   WII#        CONVERT THE INSTRUCTIONS                       CCGNC        613     A
GPO                EQ     GPO25                                                      GPO          998         I
          -FCC0675
GPO                EQ     GPO35                                                      FCC0675      135     A
GPO                                                                                 GPO          999     A
GPO        GPO26   SX6    0                                                          GPO         1000         I
          -FCC0675
GPO        GPO36   SX6    0                                                          FCC0675      136     A
GPO                SA6    L.PIT       L.PIT = 0                                      GPO         1001     A
GPO                SA6    L.TXT       L.TXT = 0                                      GPO         1002     A
GPO                SA6    L.BLK       L.BLK = 0                                      CCG1          56     A
GPO                SA6    L.TET       L.TET = 0                                      CCG1          57     A
GPO                                                                                 GPO         1003     A
GPO       *        COMPUTE *FL* USED AND ISSUE A DAYFILE MESSAGE IF > 50K .          GPO         1004         I
          -FCC0448
GPO                                                                                 GPO         1005         I
          -FCC0448
GPO       *        MU = MAX( 8*M.PL , MU )                                           GPO         1006         I
          -FCC0448
GPO       *        P2FL = MAX( P2.FL , LM + MU )                                     GPO         1007         I
          -FCC0448
GPO       *        FLU = P2.FL + FL-LWAWORK ( FL = CP.NFLS )                         GPO         1008         I
          -FCC0448
GPO       *        IF( FLU @ 50K ) RETURN                                            GPO         1009         I
          -FCC0448
GPO       *        FLU = FLU + 1077B / 100B * 100B                                   GPO         1010         I
          -FCC0448
GPO       *        BCDN = COD( FLU )                                                 GPO         1011         I
          -FCC0448
GPO       *        MESSAGE  NAMEXXX,OPT2 FL = NNNNNB *                               GPO         1012         I
          -FCC0448
GPO                                                                                 GPO         1013         I
          -FCC0448
GPO                EQ     GPO                                                        GPO         1014         I
          -CCGNC
GPO                EQ     CG$GPO                                                     CCGNC         614     A
GPO        IGP     TITLE  GRAPH PROCESSING SUBROUTINES                               GPO         1015     A
GPO       **       IGP - INITIALIZE GRAPH PROCESSING                                 GPO         1016     A
GPO       *                                                                          GPO         1017     A
GPO       *        ENTRY  (GP) = GRAPH POINTER, FWA OF GRAPH TABLE                   GPO         1018     A
GPO       *                                                                          GPO         1019     A
GPO       *        EXIT   (X6) = N.LP , N. LOOPS IN GRAPH                            GPO         1020     A

          0        1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                                                                                GPO        1021    A
GPO      IGP      ROUTINE                                                          GPO        1022    A
GPO               SA1    GP                                                        GPO        1023    A
GPO               SA2    X1+B1       GB = [GP] + 1;  GBW = [GB]                    GPO        1024    A
GPO               SX6    A2+B1                                                     GPO        1025    A
GPO               SA6    EIT         EIT = [GP] + 2  */ EDGE INDEX TABLE           GPO        1026    A
GPO                                                                                GPO        1027    A
GPO      *        BUMP PRED COUNT OF PROGRAM EXIT NODE SO IT DOESNT BECOME A       GPO        1028    A
GPO      *        INSERTION POINT FOR POST STORE CODE ( FXI ).                     GPO        1029    A
GPO                                                                                GPO        1030    A
GPO               SX5    B1                                                        GPO        1031    A
GPO               SA4    A2+B1                                                     GPO        1032    A
GPO               LX5    EI.NPP                                                    GPO        1033    A
GPO               IX7    X5+X4       NP[EIT] = NP[EIT] + 1                         GPO        1034    A
GPO               SA7    A4                                                        GPO        1035    A
GPO                                                                                GPO        1036    A
GPO               LX2    -GB.ILIP                                                  GPO        1037    A
GPO               SX3    X2                                                        GPO        1038    A
GPO               IX7    X6+X3                                                     GPO        1039    A
GPO               SB3    X7                                                        GPO        1040    A
GPO               SA7    IBA         IBA = EIT + ILI[GBW]                          GPO        1041    A
GPO               LX2    GB.ILIP-GB.NLPP                                           GPO        1042    A
GPO               SX6    X2                                                        GPO        1043    A
GPO               SA6    N.LP        N.LP = NLP[GBW]                               GPO        1044    A
GPO               MX7    0                                                         GPO        1045    A
GPO               SA7    NN          NN = 0                                        GPO        1046    A
GPO                                                                                GPO        1047    A
GPO      *        SETUP *BTT* - *BN* TO *BI*  TRANSITION TABLE                     GPO        1048         I
         -FCC0392
GPO                                                                                GPO        1049         I
         -FCC0392
GPO               SA5    B3          IP = IBA                                      GPO        1050         I
         -FCC0392
GPO               SA1    O.BTT                                                     GPO        1051         I
         -FCC0392
GPO               SB7    X1                                                        GPO        1052         I
         -FCC0392
GPO               LX5    -IP.NODEP                                                 GPO        1053         I
         -FCC0392
GPO               SB2    X5          N = NODE[IP]                                  GPO        1054         I
         -FCC0392
GPO               SB5    IM.BIP                                                    GPO        1055         I
         -FCC0392
GPO               SA7    B7          BTT(0) = 0                                    GPO        1056         I
         -FCC0392
GPO                                                                                GPO        1057         I
         -FCC0392
GPO       IGP1    SA5    A5+B1       IP = IP + 1                                   GPO        1058         I
         -FCC0392
GPO               SB2    B2-B1       N = N - 1                                     GPO        1059         I
         -FCC0392
GPO               AX4    B5,X5                                                     GPO        1060         I

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
          -FCC0392
GPO                LX5    -IM.BNP                                                GPO        1061        I
          -FCC0392
GPO                SX7    X4                                                     GPO        1062        I
          -FCC0392
GPO                SA7    B7+X5       BTT(BN[IP]) = BI[IP]                       GPO        1063        I
          -FCC0392
GPO                NZ     B2,IGP1     IF N " 0      */ NOT END OF INTERVAL LIST  GPO        1064        I
          -FCC0392
GPO                SA5    A5+B1                                                  GPO        1065        I
          -FCC0392
GPO                LX5    -IP.NODEP                                              GPO        1066        I
          -FCC0392
GPO                SB2    X5          N = NODE[IP]                               GPO        1067        I
          -FCC0392
GPO                NZ     X5,IGP1     IF N " 0                                   GPO        1068        I
          -FCC0392
GPO                                                                              GPO        1069        I
          -FCC0392
GPO       *        SETUP *BTT* AND *HNT*, SET *HN* BIT FOR HEADER NODES.         FCC0392      29     A
GPO       *        SETUP BVI FOR *HB"S* SO AS TO SIMPLIFY *FXI* LOGIC.           FCC0392      30     A
GPO       *        BTT - NODE NUMBER TO *BI* TRANSITION TABLE                    FCC0392      31     A
GPO       *        HNT - HEADER NODE TO *HB* SEARCH TABLE.                       FCC0392      32     A
GPO                                                                              FCC0392      33     A
GPO                SA0    X6                                                     FCC0392      34     A
GPO                SA1    O.BTT                                                  FCC0392      35     A
GPO                SA3    O.HNT                                                  FCC0392      36     A
GPO                SA2    O.BIT                                                  FCC0392      37     A
GPO                SB7    X1                       (B7) = O.BTT                  FCC0392      38     A
GPO                SB6    X3          HI = O.HNT                                 FCC0392      39     A
GPO                SB5    IM.BIP                                                 FCC0392      40     A
GPO                SB4    X2                       (B4) = O.BIT                  FCC0392      41     A
GPO                MX0    -IP.NODEL                                              FCC0392      42     A
GPO                SA5    B3          IP = IBA                                   FCC0392      43     A
GPO                MX7    0                                                      FCC0392      44     A
GPO                SA7    B7          BTT(0) = 0                                 FCC0392      45     A
GPO                                                                              FCC0392      46     A
GPO       IGP1     BX1    X0*X5       HNI = (NIP,PB)[IP]                         FCC0392      47     A
GPO                LX5    -IP.NODEP                                              FCC0392      48     A
GPO                SB2    X5          N = NODE[IP]                               FCC0392      49     A
GPO                SA5    A5+B1       IP = IP + 1   */ HEADER NODE               FCC0392      50     A
GPO                BX2    X5          IPF = [IP]                                 FCC0392      51     A
GPO                                                                              FCC0392      52     A
GPO       IGP2     AX4    B5,X5                                                  FCC0392      53     A
GPO                LX5    -IM.BNP                                                FCC0392      54     A
GPO                SX7    X4                                                     FCC0392      55     A
GPO                SA7    B7+X5       BTT(BN[IP]) = BI[IP]                       FCC0392      56     A
GPO                SB2    B2-B1       N = N - 1                                  FCC0392      57     A
GPO                SA5    A5+B1       IP = IP + 1                                FCC0392      58     A
GPO                NZ     B2,IGP2     IF N " 0                                   FCC0392      59     A
GPO                                                                              FCC0392      60     A
GPO                LX2    59-IM.SCRP                                             FCC0392      61     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                PL     X2,IGP3     IF ^SCR[IPF]  */ HEADER ^ IN SCR                FCC0392      62     A
GPO                LX2    1+IM.SCRP-IM.BIP  I = BI[IPF]                               FCC0392      63     A
GPO                SX2    X2                                                          FCC0392      64     A
GPO                BX7    X1+X2                                                       FCC0392      65     A
GPO                SA7    B6          [HI] = HNW(HNI,I)                               FCC0392      66     A
GPO                SB6    B6+B1       HI = HI + 1                                     FCC0392      67     A
GPO                                                                                  FCC0392      68     A
GPO                LX1    -IP.PBP+1   HBI = 2*PB[HNI]                                 FCC0392      69     A
GPO                SX4    B1                                                          FCC0392      70     A
GPO                SA3    B4+X2                                                       FCC0392      71     A
GPO                LX4    BI.HNP                                                      FCC0392      72     A
GPO                BX6    X4+X3       HN[BIT(I)] = 1                                  FCC0392      73     A
GPO                SA6    A3                                                          FCC0392      74     A
GPO                LX3    -BI.BVIP                                                    FCC0392      75     A
GPO                SX7    X3                                                          FCC0392      76     A
GPO                LX7    BI.BVIP                                                     FCC0392      77     A
GPO                SA7    B4+X1       BVI[BIT(HBI)] = BVI[BIT(I)]                     FCC0392      78     A
GPO                                                                                  FCC0392      79     A
GPO       IGP3     NZ     X5,IGP1     IF [IP] " 0   */ ^ END OF INTERVAL LIST         FCC0392      80     A
GPO                                                                                  FCC0392      81     A
GPO                SX6    A0                                                          FCC0392      82     A
GPO                SX7    B6                                                          FCC0392      83     A
GPO                SA7    L.HNT       L.HNT = HI    */ LWA+1 OF HNT                   FCC0392      84     A
GPO                EQ     IGP                                                         GPO        1070     A
GPO       FNL      SPACE  3,14        FNL                                             GPO        1071     A
GPO      **        FNL - FIND NEXT LOOP ( STRONGLY CONNECTED REGION  )                GPO        1072     A
GPO      *                                                                           GPO        1073     A
GPO      *         EXIT   (IBA) = INTERVAL LIST ADDRESS FOR *SCR*                     GPO        1074     A
GPO      *                (NN) = N.NODES IN INTERVAL                                 GPO        1075     A
GPO      *                (X4) = IM = [IBA] , FIRST INTERVAL MEMBER WORD             GPO        1076     A
GPO                                                                                  GPO        1077     A
GPO       FNL0     SX6    A4                                                          GPO        1078     A
GPO                SA6    A1          IBA = IM                                        GPO        1079     A
GPO                SX7    B2                                                          GPO        1080     A
GPO                SA7    A2          NN = NODE[IP]                                   GPO        1081     A
GPO                LX4    1+IM.SCRP                                                   GPO        1082     A
GPO                                                                                  GPO        1083     A
GPO       FNL      ROUTINE                                                            GPO        1084     A
GPO                SA1    IBA                                                         GPO        1085     A
GPO                SA2    A1+B1                                                       GPO        1086     A
GPO                IX7    X1+X2                                                       GPO        1087     A
GPO                SA5    X7          IP = IBA + NN */ POINT TO IP WORD OF NEXT       GPO        1088     A
GPO                                                                                  GPO        1089     A
GPO       FNL1     SA4    A5+B1       IM = IP + 1                                     GPO        1090     A
GPO                LX5    -IP.NODEP                                                   GPO        1091     A
GPO                SB2    X5          N = NODE[IP]                                    GPO        1092     A
GPO                LX4    59-IM.SCRP                                                  GPO        1093     A
GPO                MI     X4,FNL0     IF SCR[IM]    */ HEADER IN A *SCR*              GPO        1094     A
GPO                SA5    A4+B2       IP = IM + N   */ ADVANCE TO NEXT HEADER         GPO        1095     A
GPO                EQ     FNL1                                                        GPO        1096     A
GPO       GNG      SPACE  2,14        GNG                                             GPO        1097     A
GPO      **        GNG - GET NEXT GRAPH TABLE                                         GPO        1098     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO      *                                                                  GPO        1099    A
GPO      *         EXIT    (X6) = 0 IF END OF GRAPHS, ELSE (GP) = FWA OF GRAPH  GPO   1100    A
GPO                                                                         GPO        1101    A
GPO       GNG0     SA2     GP                                               GPO        1102    A
GPO                SA3     X2                                               GPO        1103    A
GPO                LX3     -GA.LFP                                          GPO        1104    A
GPO                SX6     X3                                               GPO        1105    A
GPO                ZR      X6,GNG      IF LF[GP] = 0 */ LEAVE *GP* ALONE    GPO        1106    A
GPO                IX7     X2+X6       GP = GP + LF[GP]                     GPO        1107    A
GPO                SA7     A2                                               GPO        1108    A
GPO       GNG      ROUTINE                                                  GPO        1109    A
GPO                SA1     GIC                                              GPO        1110    A
GPO                ZR      X1,GNG0     IF GIC = 0    */ ALL GRAPHS ARE INCORE GPO      1111    A
GPO                                                                         GPO        1112    A
GPO                SA4     X1+B1       RAW = [GIC+1] */ RANDOM INDEX WORD   GPO        1113    A
GPO                SX7     X1+B1       GIC = GIC + 1                        GPO        1114    A
GPO                MX6     0                                                GPO        1115    A
GPO                SA7     A1                                               GPO        1116    A
GPO                ZR      X4,GNG      IF RAW = 0    */ END OF GRAPHS       GPO        1117    A
GPO                SA2     GP                                               GPO        1118    A
GPO                LX4     -RI.LENP                                         GPO        1119    A
GPO                SX5     X4          L = LEN[RAW]                         GPO        1120    A
GPO                LX4     RI.LENP                                          GPO        1121    A
GPO                CALL    GBD         GBD( FSCM = GP , RAW , L )           GPO        1122        I
          -CCGNC
GPO                CALL    GBD#        GBD( FSCM = GP , RAW , L )           CCGNC       615    A
GPO                SX6     B1                                               GPO        1123    A
GPO                EQ      GNG                                              GPO        1124    A
GPO       IRP      TITLE   IRP - INITIALIZE REGION PROCESSING               GPO        1125    A
GPO      **        IRP - INITIALIZE REGION PROCESSING FOR A *SCR*           GPO        1126    A
GPO                                                                         GPO        1127    A
GPO       IRP      ROUTINE                                                  GPO        1128    A
GPO                SA5     NN                                               GPO        1129    A
GPO                LX4     B1,X5                                            GPO        1130    A
GPO                ALLOC   BST,X4+3    ALLOC( BST , 2*NN+3 )                GPO        1131    A
GPO                                                                         GPO        1132    A
GPO      *         SETUP *BST* FOR PROCESSING OF *SCR*                      GPO        1133    A
GPO      *         INITIALIZE *BST,BIT* ENTRIES FOR HOLDING BLOCK           GPO        1134    A
GPO                                                                         GPO        1135    A
GPO                SA4     IBA                                              GPO        1136    A
GPO                SA3     O.BIT                                            GPO        1137    A
GPO                SB5     X5          N = NN                               GPO        1138    A
GPO                SA1     X4          IP = [IBA]                           GPO        1139    A
GPO                SA5     A1-B1       IHW = [IBA-1]                        GPO        1140    A
GPO                LX5     -IP.PBP+1                                        GPO        1141    A
GPO                SX6     X5          HN = 2*PB[IHW]  */ *BN* OF HOLDING BLOCK GPO   1142    A
GPO                MX7     0                                                GPO        1143    A
GPO                SA6     HBI         HBI(1) = HN                          GPO        1144    A
GPO                MX0     -BA.HBBL                                         GPO        1145    A
GPO                LX6     BA.BIP                                           GPO        1146    A
GPO                LX0     BA.HBBP                                          GPO        1147    A
GPO                BX6     -X0+X6                                           GPO        1148    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO              SA6    X2          BST(0) = BAW(HBB,HN,0,0)                GPO        1149    A
GPO              SB7    X2                                                  GPO        1150    A
GPO              SA7    X2+B1       BST(1) = 0;  I = 2                      GPO        1151    A
GPO              MX0    -IM.BIL                                             GPO        1152    A
GPO              SB3    X3+B1                                               GPO        1153    A
GPO              SB2    X5                                                  GPO        1154    A
GPO              LX0    IM.BIP                                              GPO        1155    A
GPO              SA2    X3+B2                                               GPO        1156    A
GPO              BX7    -X0*X1      BN = BI[IP]   */ BN OF HEADER           GPO        1157    A
GPO              LX7    BI.PBNP-IM.BIP                                      GPO        1158    A
GPO              BX7    X7+X2       PBN[BIT(HN)] = BN  */ LINK *HB* TO HEADER GPO      1159    A
GPO              MX6    -BI.BVIL                                            FCC0392      85    A
GPO              LX6    BI.BVIP                                             FCC0392      86    A
GPO              BX7    X6*X7       BVI[BIT(HN)] = 0                        FCC0392      87    A
GPO              SA7    A2                                                  GPO        1160    A
GPO              SB2    B0          TLEN = 0                                GPO        1161    A
GPO                                                                         GPO        1162    A
GPO       IRP1   LX1    59-IM.SCRP                                          GPO        1163    A
GPO              PL     X1,IRP2     IF ^SCR[IP]   */ BLOCK NOT IN *LOOP*    GPO        1164    A
GPO              LX1    1+IM.SCRP                                           GPO        1165    A
GPO              BX6    -X0*X1                                              GPO        1166    A
GPO              BX7    X1                                                  GPO        1167    A
GPO              LX1    -IM.BIP                                             GPO        1168    A
GPO              SA2    B3+X1       RIW = BIT(BI[IP]+1)                     GPO        1169    A
GPO              LX2    -RI.LENP                                            GPO        1170    A
GPO              SB2    B2+X2       TLEN = TLEN + LEN[RIW]                  GPO        1171    A
GPO              LX6    BA.BIP-IM.BIP                                       GPO        1172    A
GPO              SA6    A6+2        BST(I) = BAW(0,BI[IP],0,0)              GPO        1173    A
GPO              SA7    A6+1        BST(I+1) = 0;  I = I + 2                GPO        1174    A
GPO                                                                         GPO        1175    A
GPO       IRP2   SB5    B5-B1       N = N - 1                               GPO        1176    A
GPO              SA1    A1+B1       IP = IP + 1                             GPO        1177    A
GPO              NZ     B5,IRP1     IF N " 0                                GPO        1178    A
GPO                                                                         GPO        1179    A
GPO              MX6    0                                                   GPO        1180    A
GPO              SA6    A7+B1       BST(I) = 0    */ TABLE TERMINATOR       GPO        1181    A
GPO              SB6    A6+B1                                               GPO        1182    A
GPO              SX7    B6-B7       L.BST = I + 1                           GPO        1183    A
GPO              SA7    L.BST                                               GPO        1184    A
GPO       IRP    SPACE  1,10                                                GPO        1185    A
GPO      *       INITIALIZE FLAGS, CLEAR REGION BIT VECTORS                 GPO        1186    A
GPO                                                                         GPO        1187    A
GPO              SA1    L.SBV                                               GPO        1188    A
GPO              SA2    DVA                                                 GPO        1189    A
GPO              SETZERO X2,X1      SETZERO( [DVA] , L.SBV )                GPO        1190    A
GPO              MX5    -24                                                 GPO        1191    A
GPO              BX7    -X5         MRA = ^MASK(36)                         GPO        1192    A
GPO              SA7    MRA                                                 GPO        1193    A
GPO              SA7    MRA.X       MRA.X = MRA                             FCC1254      10    A
GPO              SETZERO N.HB,N.CLC SETZERO( N.HB , N.CLC )                 GPO        1194    A
GPO              SA1    L.TET                                               GPO        1195    A
GPO              MX6    0                                                   GPO        1196    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO              BX7    X1                                                GPO        1197    A
GPO              SA7    ITL        ITL = L.TET                            GPO        1198    A
GPO              SA7    TTI        TTI = L.TET                            GPO        1199    A
GPO              SA6    NTP        NTP = 0                                GPO        1200    A
GPO              SA6    LGL        LGL = 0                                GPO        1201    A
GPO              SA6    UXR+1      UXR(2) = 0                             GPO        1202    A
GPO              SA6    NBIP       NBIP = 0                               FCC0514      30    A
GPO                                                                       GPO        1203    A
GPO      *       GET HEADER NODE, SET LINENR OF *BOS*, LOOP LABEL, ETC.   GPO        1204    A
GPO                                                                       GPO        1205    A
GPO              SX7    2                                                 GPO        1206    A
GPO              SA7    BSI        BSI = 2                                GPO        1207    A
GPO              CALL   RTB        READ FIRST BLOCK INTO *BLK*            GPO        1208         I
         -CCGNC
GPO              CALL   RTB#       READ FIRST BLOCK INTO *BLK*            CCGNC       616    A
GPO                                                                       GPO        1209    A
GPO              SA1    O.SEQ      T = [O.SEQ]                            GPO        1210    A
GPO              SA3    IBA                                               GPO        1211         I
         -CCG21
GPO              SA2    X1+B1                                             CCG21         7    A
GPO              LX2    59-R2.LONGP                                       CCG21         8    A
GPO              PL     X2,IRP3    IF ^LONG[R2]  */ NOT DO=LONG           CCG21         9    A
GPO              MX5    -16                                               CCG21        10    A
GPO              LX5    8                                                 CCG21        11    A
GPO              BX7    -X5        MRA = 77777400B  */ NO B-REGS AVAIL    CCG21        12    A
GPO              SA7    MRA                                               CCG21        13    A
GPO              SA7    MRA.X      MRA.X = MRA                            CCG21        14    A
GPO       IRP3   SA3    IBA                                               CCG21        15    A
GPO              SA5    X1+5                                              GPO        1212    A
GPO              SA4    X1                                                GPO        1213    A
GPO              LX7    X5         LIH(2) = [T+5]  */ R2 WORD OF LOOP *LAB*  GPO      1214    A
GPO              SA5    A5-B1      R1 = [T+4]      */ R1 WORD OF *LAB*    GPO        1215    A
GPO              BX6    X4                                                GPO        1216    A
GPO              SA6    HBI+1      HBI(2) = [T]  */ R1 WORD OF *BOS*      GPO        1217    A
GPO              SA7    LIH+1                                             GPO        1218    A
GPO              LX6    -R1.INP                                           GPO        1219    A
GPO              SX6    X6         LINENR = IN[T]  */ LINE NUMBER OF LOOP GPO        1220         I
         -CCG1
GPO              SX6    X6         CO.CSN  = IN[T]  */ LINE NUMBER OF LOOP CCG1          58         I
         -CCGNC
GPO              SX6    X6         HO$CSN  = IN[T]  */ LINE NUMBER OF LOOP CCGNC       617    A
GPO              SA6    =XLINENR                                          GPO        1221         I
         -CCG1
GPO              SA6    =XCO.CSN                                          CCG1          59         I
         -CCGNC
GPO              SA6    =XHO$CSN                                          CCGNC       618    A
GPO              SX7    X5                                                GPO        1222    A
GPO              SA7    A7-B1      LIH(1) = IH[R1]                        GPO        1223    A
GPO              MX7    0                                                 CCGA021      35    A
GPO              SA7    A5+B1      [T+5] = 0     */ CLEAR R2 WORD OF LABEL DEF  CCGA021  36    A
GPO              SA3    X3-1       IPW = [IBA-1]                          GPO        1224    A
GPO              LX3    59-IP.GLP                                         GPO        1225    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                PL     X3,IRP      IF ^GL[IPW]   */ NO LABEL TRANSFER NECESSAR        GPO         1226     A
GPO                                                                                      FCC0408       31     A
GPO                LX3    IP.GLP-IP.FTP                                                  FCC0408       32     A
GPO                BX6    X3                                                             FCC0408       33     A
GPO                SA6    FTH         FTH = FT[IPW] */ SAVE FOR *TRP*                    FCC0408       34     A
GPO        IRP     SPACE  1,10                                                           GPO         1227     A
GPO       *        REGION HAS MORE THAN 1 PRED, SETUP *LGL* TO INDICATE CREATION         GPO         1228     A
GPO       *        OF A GENERATED LABEL FOR THE TRANSFERS INSIDE THE LOOP.               GPO         1229     A
GPO                                                                                      GPO         1230     A
GPO                SA4    N.GL0                                                          GPO         1231     A
GPO                UX5    B2,X5                                                          GPO         1232     A
GPO                SX6    X4+B1       N.GL0 = N.GL0 + 1                                  GPO         1233     A
GPO                SA6    A4                                                             GPO         1234     A
GPO                SX5    X5          LL = IH[R1]                                        GPO         1235     A
GPO                SX6    X6+I.GL     GIH = I.GL + N.GL0                                 GPO         1236     A
GPO                IX2    X6-X5                                                          GPO         1237     A
GPO                LX2    18                                                             GPO         1238     A
GPO                BX7    X2+X5       LGL = SHIFT(GIH-LL,18) ! LL                        GPO         1239     A
GPO                PX6    B2,X6                                                          GPO         1240     A
GPO                SA7    LGL                                                            GPO         1241     A
GPO                SA6    X1+4        [T+4] = TYIV(OC.LAB,0,GIH)                         GPO         1242     A
GPO                                                                                      GPO         1243     A
GPO                SA2    HBI                                                            GPO         1244     A
GPO                SA1    O.BIT                                                          GPO         1245     A
GPO                SX0    B1                                                             GPO         1246     A
GPO                SB3    X2                                                             GPO         1247     A
GPO                SA3    X1+B3                                                          GPO         1248     A
GPO                LX0    BI.ILP                                                         GPO         1249     A
GPO                BX6    X0+X3       IL[BIT(HN)] = 1  */ SET *INIT* LAB BIT             GPO         1250     A
GPO                SA6    A3                                                             GPO         1251     A
GPO                EQ     IRP                                                            GPO         1252          I
          -FCC1319
GPO                                                                                      FCC1319       26     A
GPO       *        SETUP INFO IN SYMTAB AND *LCT* IN CASE THERE ARE TRANSFERS            FCC1319       27     A
GPO       *        FROM LOOPS INSIDE THIS LOOP TO THE HEAD OF IT THAT WILL BE            FCC1319       28     A
GPO       *        ADJUSTED IN *POST*.                                                   FCC1319       29     A
GPO                                                                                      FCC1319       30     A
GPO                SA3    N.GL0                                                          FCC1319       31     A
GPO                LX3    LC.GLNP                                                        FCC1319       32     A
GPO                BX6    X3+X2                                                          FCC1319       33     A
GPO                SA4    L.LCT       L = L.LCT                                          FCC1319       34     A
GPO                SA3    O.LCT                                                          FCC1319       35     A
GPO                SB3    X3                                                             FCC1319       36     A
GPO                SA6    B3+X4       LCT(L) = LC(0,N.GL0,HBI)                           FCC1319       37     A
GPO                SX7    X4+B1       L.LCT = L.LCT + 1                                  FCC1319       38     A
GPO                SA7    A4                                                             FCC1319       39     A
GPO                SA1    O.SYM                                                          FCC1319       40          I
          -CCG1
GPO                LX5    1                                                              FCC1319       41          I
          -CCG1
GPO                IX7    X1-X5                                                          FCC1319       42          I
          -CCG1

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                 SA5    X7          WORDA = SYM(IH)                          FCC1319      43          I
          -CCG1
GPO                 SX0    B1                                                   FCC1319      44          I
          -CCG1
GPO                 LX0    P.LC                                                 FCC1319      45          I
          -CCG1
GPO                 BX7    X0+X5                                                FCC1319      46          I
          -CCG1
GPO                 MX0    -L.LCO                                               FCC1319      47          I
          -CCG1
GPO                 SA1    A5-B1       WORDB = WORDA-1                          FCC1319      48          I
          -CCG1
GPO                 LX5    59-P.LC                                              FCC1319      49          I
          -CCG1
GPO                 PL     X5,IRP4     IF ^LC[WORDA] */ FIRST TIME              FCC1319      50          I
          -CCG1
GPO                                                                             FCC1319      51          I
          -CCG1
GPO                 BX7    -X0*X1      J = LCO[WORDB]                           FCC1319      52          I
          -CCG1
GPO                 SA1    =XO.SYM                                              CCG1         60     A
GPO                 LX7    B1,X5                                                CCG1         61     A
GPO                 IX5    X7+X5                                                CCG1         62     A
GPO                 SB4    X1+B1                                                CCG1         63     A
GPO                 SA5    B4+X5       WORDB = SYM(3*IH+1)                      CCG1         64     A
GPO                 SX0    B1                                                   CCG1         65     A
GPO                 LX0    WB.LCP                                               CCG1         66     A
GPO                 BX7    X0+X5                                                CCG1         67     A
GPO                 MX0    -WA.LCOL                                             CCG1         68     A
GPO                 SA1    A5-B1       WORDA = SYM(3*IH)                        CCG1         69     A
GPO                 LX5    59-WB.LCP                                            CCG1         70     A
GPO                 PL     X5,IRP4     IF ^LC[WORDB] */ FIRST TIME              CCG1         71     A
GPO                                                                             CCG1         72     A
GPO                 BX7    -X0*X1      J = LCO[WORDA]                           CCG1         73     A
GPO       +         SA2    B3+X7       JL = J                                   FCC1319      53     A
GPO                 LX2    -LC.NEXTP                                            FCC1319      54     A
GPO                 SX7    X2+         J = NEXT[LCT(JL)]                        FCC1319      55     A
GPO                 NZ     X7,*-1      IF J " 0                                 FCC1319      56     A
GPO                                                                             FCC1319      57     A
GPO                 BX6    X4+X2       NEXT[LCT(JL)] = L.LCT - 1                FCC1319      58     A
GPO                 LX6    LC.NEXTP                                             FCC1319      59     A
GPO                 SA6    A2                                                   FCC1319      60     A
GPO                 EQ     IRP5                                                 FCC1319      61     A
GPO                                                                             FCC1319      62     A
GPO        IRP4     SA7    A5          LC[WORDA] = 1                            FCC1319      63          I
          -CCG1
GPO                 BX1    X0*X1                                                FCC1319      64          I
          -CCG1
GPO                 BX7    X4+X1       LCO[WORDB] = L                           FCC1319      65          I
          -CCG1
GPO        IRP4     SA7    A5          LC[WORDB] = 1                            CCG1         74     A
GPO                 BX1    X0*X1                                                CCG1         75     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                BX7    X4+X1       LCO[WORDA] = L                          CCG1          76     A
GPO                SA7    A1                                                  FCC1319       66     A
GPO                                                                           FCC1319       67     A
GPO       IRP5     EQ     IRP                                                 FCC1319       68     A
GPO       IHB      SPACE  3,14        IHB                                     GPO         1253     A
GPO      **        IHB - INITIALIZE HOLDING BLOCK                             GPO         1254     A
GPO                                                                           GPO         1255     A
GPO       IHB      ROUTINE                                                    GPO         1256     A
GPO                SA2    HBI         HN = HBI(1)                             GPO         1257     A
GPO                SA1    A2+B1                                               GPO         1258     A
GPO                MX0    -18                                                 GPO         1259     A
GPO                SA4    O.TXT       R1 = [O.TXT]                            GPO         1260     A
GPO                SA5    LGL                                                 GPO         1261     A
GPO                BX3    X0*X1                                               GPO         1262     A
GPO                IX6    X3+X2       [R1] = TYIV(OC.BOS,LINENR,HN)           GPO         1263          I
          -CCG1
GPO                IX6    X3+X2       [R1] = TYIV(OC.BOS,CO.CSN ,HN)          CCG1          77          I
          -CCGNC
GPO                IX6    X3+X2       [R1] = TYIV(OC.BOS,HO$CSN ,HN)          CCGNC        619     A
GPO                SA6    X4+                                                 GPO         1264     A
GPO                ZR     X5,IHB      IF LGL = 0    */ NO GENERATED LABEL     GPO         1265     A
GPO                                                                           GPO         1266     A
GPO                ALLOC  TXT,4       ALLOC( TXT , 4 )                        GPO         1267     A
GPO                SB7    X2                                                  GPO         1268     A
GPO                SB2    OC.LAB                                              GPO         1269     A
GPO                SX4    X5          LL = SETX(LGL)  */ CURRENT LOOP LABEL   GPO         1270     A
GPO                AX5    18                                                  GPO         1271     A
GPO                IX6    X4+X5       LIH = GLB[LGL]                          GPO         1272          I
          -CCGA021
GPO                SA6    LIH                                                 GPO         1273          I
          -CCGA021
GPO                MX7    0                                                   GPO         1274          I
          -CCGA021
GPO                IX7    X4+X5       LIH = GLB[LGL]  */ SAVE GEN LABEL IN R2 CCGA021        37     A
GPO                SA7    LIH         UPDATE LIH                              CCGA021        38     A
GPO                SX6    X4                                                  GPO         1275     A
GPO                RJ     SRI         SRI(  LAB  0,LL )  */ MOVE LABEL OUT    GPO         1276     A
GPO                EQ     IHB                                                 GPO         1277     A
GPO       TRP      TITLE  TRP - TERMINATE REGION PROCESSING                   GPO         1278     A
GPO      **        TRP - TERMINATE REGION PROCESSING FOR A LOOP               GPO         1279     A
GPO                                                                           GPO         1280     A
GPO       TRP      ROUTINE                                                    GPO         1281     A
GPO                                                                           GPO         1282          I
          -CCG1
GPO      *         CHECK LOOP CONTROL VAR AND SET *M* BIT IN SYMTAB IF IT IS ^RD GPO       1283          I
          -CCG1
GPO                                                                           GPO         1284          I
          -CCG1
GPO                SA4    LIH+1                                               GPO         1285          I
          -CCG1
GPO                ZR     X4,TRP1     IF LIH(2) = 0 */ NOT A DO LOOP          GPO         1286          I
          -CCG1

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                 SA1    O.UDT                                                GPO        1287          I
          -CCG1
GPO                 LX4    -IH.ICVP                                             GPO        1288          I
          -CCG1
GPO                 SB4    X4                                                   GPO        1289          I
          -CCG1
GPO                 SA3    X1+B4       UI = UDT(ICV[LIH])                       GPO        1290          I
          -CCG1
GPO                 LX3    59-UD.^RDP                                           GPO        1291          I
          -CCG1
GPO                 PL     X3,TRP1     IF RD[UI]                                GPO        1292          I
          -CCG1
GPO                 SA4    A4-B1       IH = LIH(1)   */ *IH* OF )XX             GPO        1293          I
          -CCG1
GPO                 SA5    O.SYM                                                GPO        1294          I
          -CCG1
GPO                 SX7    B1                                                   GPO        1295          I
          -CCG1
GPO                 LX4    1                                                    GPO        1296          I
          -CCG1
GPO                 IX5    X5-X4                                                GPO        1297          I
          -CCG1
GPO                 SA3    X5-1        WORDB = [O.SYM-2*IH-1]                   GPO        1298          I
          -CCG1
GPO                 LX7    P.FLG+OF.MP                                          GPO        1299          I
          -CCG1
GPO                 BX6    X3+X7       MAT[WORDB] = 1  */ SET MATERIALIZATION FLAG  GPO    1300          I
          -CCG1
GPO                 SA6    A3                                                   GPO        1301          I
          -CCG1
GPO                                                                             GPO        1302          I
          -CCG1
GPO        TRP1     SA4    L.BST                                                GPO        1303          I
          -CCG1
GPO                 SA4    L.BST                                                CCG1         78     A
GPO                 SX5    X4-5                                                 GPO        1304     A
GPO                 RJ     CRB         RESET *UDT*                              GPO        1305     A
GPO                 SX6    0                                                    GPO        1306     A
GPO                 SA6    L.IIT       L.IIT = 0                                GPO        1307     A
GPO                 SA6    BSI         BSI = 0                                  GPO        1308     A
GPO                 SA6    BBV         BBV = 0                                  GPO        1309     A
GPO                                                                             GPO        1310     A
GPO                 CALL   RTB         GET HOLDING BLOCK                        GPO        1311          I
          -CCGNC
GPO                 CALL   RTB#        GET HOLDING BLOCK                        CCGNC       620     A
GPO                 SA4    HBI                                                  GPO        1312     A
GPO                 SA5    O.SEQ                                                GPO        1313     A
GPO                 MX7    -UD.STCP                                             GPO        1314     A
GPO                 BX6    X4                                                   GPO        1315     A
GPO                 SA7    BCM         BCM = *REGION CLEAR MASK*                GPO        1316     A
GPO                 SA6    BN          BN = HBI(1)                              GPO        1317     A
GPO                 CALL   FUD         FORM BIT VECTORS FOR *HB*                GPO        1318     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                SX6    1                                                    GPO        1319     A
GPO                SA6    BBV          BBV = 1                                 GPO        1320     A
GPO                MX7    UD.MVL                                               GPO        1321     A
GPO                LX7    UD.MVL+UD.MVP                                        GPO        1322     A
GPO                SA7    BCM          BCM = *MV*                              GPO        1323     A
GPO                                                                            GPO        1324     A
GPO      *         *HB* LIVE EXIT VECTOR IS REGION LIVE ENTRY VECTOR           GPO        1325     A
GPO                                                                            GPO        1326     A
GPO                SA1    VL                                                   GPO        1327     A
GPO                SA2    O.BVT                                                GPO        1328     A
GPO                SA3    L.BVT                                                GPO        1329     A
GPO                SA4    LEA                                                  GPO        1330     A
GPO                IX7    X3-X1                                                GPO        1331     A
GPO                SB4    X2                        (B4) = [O.BVT]             GPO        1332     A
GPO                SB5    B4+X7        LVA = O.BVT + L.BVT - VL                GPO        1333     A
GPO                SB2    B0                                                   GPO        1334     A
GPO                SB3    X1                                                   GPO        1335     A
GPO       TRP2     SA5    X4+B2        MOVE( VL , [LEA] , LVA )                GPO        1336     A
GPO                BX6    X5                                                   GPO        1337     A
GPO                SA6    B5+B2                                                GPO        1338     A
GPO                SB2    B2+B1                                                GPO        1339     A
GPO                LT     B2,B3,TRP2                                           GPO        1340     A
GPO                                                                            GPO        1341     A
GPO      *         SAVE REGION BIT VECTORS ( DEF, LU, USE, LX ) IN THE HEADER  GPO        1342     A
GPO      *         NODE BIT VECTOR SLOT.                                       GPO        1343     A
GPO                                                                            GPO        1344     A
GPO                SA4    O.BIT                                                GPO        1345     A
GPO                SA5    HBI+1        BN = HBI(2)   */ BN OF HEADER           GPO        1346     A
GPO                SA2    DVA                                                  GPO        1347     A
GPO                LX1    2                                                    GPO        1348     A
GPO                SB6    X4                                                   GPO        1349     A
GPO                SA3    B6+X5        BIW = BIT(BN)                           GPO        1350     A
GPO                LX3    -BI.BVIP                                             GPO        1351     A
GPO                MOVE   X1,X2,B4+X3 MOVE( 4*VL , [DVA] , O.BVT + BVI[BIW] )  GPO        1352     A
GPO                                                                            GPO        1353     A
GPO                SA5    O.SEQ                                                GPO        1354          I
          -FCC0408
GPO                SA1    FTH                                                  FCC0408      35     A
GPO                PL     X1,TRP3      IF ^FTH       */ NOT A FALL THROUGH TO *HN*  FCC0408  36          I
          -FCC0675
GPO                MI     X1,TRP3      IF FTH < 0    */ FALL THROUGH TO *HN*    FCC0675     137     A
GPO                SA3    B6+X5        BIW = BIT(HN)                           FCC0675     138     A
GPO                LX3    59-BI.RBP                                            FCC0675     139     A
GPO                MI     X3,TRP4      IF RB[BIW]    */ PROGRAM BLOCK           FCC0675     140     A
GPO                LX3    BI.RBP-BI.FTHP                                       FCC0675     141     A
GPO                PL     X3,TRP4      IF ^FTH[BIW]  */ INNER LP NOT *FTH*      FCC0675     142     A
GPO                                                                            FCC0408      37     A
GPO      *         LOOP CONTAINS A FALL THROUGH TO THE HEADER NODE, AND REMOTE FCC0408      38     A
GPO      *         ENTRY BY A JUMP, SET *FJ* IN *BIT* AND APPEND A FINAL *UJP* FCC0408      39     A
GPO      *         TO THE *HB*.                                                FCC0408      40     A
GPO                                                                            FCC0408      41     A
GPO                MX7    0                                                    FCC0408      42          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
          -FCC0675
GPO                 SA7    A1          FTH = 0                                  FCC0408      43          I
          -FCC0675
GPO                 SA2    L.SEQ                                                FCC0408      44          I
          -FCC0675
GPO                 SB2    X2-12                                                FCC0408      45          I
          -FCC0675
GPO                 LE     B2,TRP3     IF L.SEQ @ 12 */ NOTHING MOVED FROM THE LOO  FCC0408  46        I
          -FCC0675
GPO                 SA1    HBI                                                  FCC0408      47          I
          -FCC0675
GPO                 SA2    O.BIT                                                FCC0408      48          I
          -FCC0675
GPO                 SX0    B1                                                   FCC0408      49          I
          -FCC0675
GPO                 IX7    X1+X2                                                FCC0408      50          I
          -FCC0675
GPO                 LX0    BI.FJP                                               FCC0408      51          I
          -FCC0675
GPO                 SA3    X7                                                   FCC0408      52          I
          -FCC0675
GPO                 BX6    X0+X3       FJ[BIT(HBI)] = 1                         FCC0408      53          I
          -FCC0675
GPO        TRP3     MX7    0           FTH = 0                                  FCC0675     143     A
GPO                 SA7    A1                                                   FCC0675     144     A
GPO                 SA1    HBI                                                  FCC0675     145     A
GPO                 SA2    L.SEQ                                                FCC0675     146         I
          -FCC2091
GPO                 SA3    B6+X1                                                FCC0675     147     A
GPO                 SX0    B1                                                   FCC0675     148     A
GPO                 LX0    BI.FTHP                                              FCC0675     149     A
GPO                 BX6    X0+X3       FTH[BIT(HBI)] = 1                        FCC0675     150     A
GPO                 SA6    A3                                                   FCC0675     151     A
GPO                 SB2    X2-12                                                FCC0675     152         I
          -FCC2091
GPO                 LE     B2,TRP4     IF L.SEQ @ 12 */ NOTHING MOVED FROM LP   FCC0675     153         I
          -FCC2091
GPO                 LX0    BI.FJP-BI.FTHP                                       FCC0675     154     A
GPO                 BX6    X0+X6       FJ[BIT(HBI)] = 1                         FCC0675     155     A
GPO                 SA6    A3                                                   FCC0408      54     A
GPO                 CALL   MBT         MOVE *HB* TO *TXT*                       FCC0408      55     A
GPO                 ALLOC  TXT,4       ALLOC( TXT , 4 )                         FCC0408      56     A
GPO                 SA5    N.GL0                                                FCC0408      57     A
GPO                 SB6    X3-8        TI = L.TXT - 8                           FCC0408      58     A
GPO                 SB2    OC.UJP                                               FCC0408      59     A
GPO                 SX6    I.GL+X5                                              FCC0408      60     A
GPO                 SB7    X2          TB = O.TXT                               FCC0408      61     A
GPO                 MX7    0                                                    FCC0408      62     A
GPO                 RJ     SRI         SRI(  UJP  0,I.GL+N.GL0 )                FCC0408      63     A
GPO                 R=     B2,OC.EOQ                                            FCC0408      64     A
GPO                 MX6    0                                                    FCC0408      65     A
GPO                 RJ     SRI         SRI(  EOQ  0,0 )                         FCC0408      66     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                CALL   WTB         REWRITE THE BLOCK                          FCC0408      67     A
GPO                CALL   RTB         RESET *SEQ* POINTERS                       FCC0408      68          I
          -CCGNC
GPO                CALL   RTB#        RESET *SEQ* POINTERS                       CCGNC       621     A
GPO                                                                              FCC0408      69     A
GPO       TRP3     SA5    O.SEQ                                                 FCC0408      70          I
          -FCC0675
GPO       TRP4     SA5    O.SEQ                                                 FCC0675     156     A
GPO                SA4    X5-1        BHW = [O.SEQ-1]                           GPO        1355     A
GPO                SX0    B1                                                    GPO        1356     A
GPO                LX0    BH.PRIP                                               GPO        1357     A
GPO                BX4    -X0*X4      PRI[BHW] = 0                              GPO        1358     A
GPO                LX0    BH.BSTP+BH.BSTL-1-BH.PRIP                             GPO        1359     A
GPO                BX6    X0+X4       BST[BHW] = 1S17  */ INDICATE NOT IN *BST* GPO        1360     A
GPO                SA6    A4                                                    GPO        1361     A
GPO                RJ     CBT         CLEAR BLOCK TABLE                         GPO        1362     A
GPO                EQ     TRP                                                   GPO        1363     A
GPO                                                                              GPO        1364          I
          -FCC0408
GPO       CBT      SPACE  3,14                                                  FCC0408      71     A
GPO       **       CBT - CLEAR *BST*, RELEASE BLOCKS THAT WERE NOT CODED        GPO        1365     A
GPO       *        INSERT HB NUMBER OF OUTER LPS IN BIT ENTRIES OF INNER LPS.   FCC0675     157          I
          -FCC1319
GPO       *        INSERT *HB* NUMBER OF THIS LOOP IN *BIT* ENTRIES OF THIS     FCC1319      69     A
GPO       *        REGION. THIS INFO IS USED ON THE BACK PASS IN *POST*.        FCC1319      70     A
GPO                                                                              GPO        1366     A
GPO       CBT0     SX6    0                                                     GPO        1367     A
GPO                SA6    L.BST       L.BST = 0                                 GPO        1368     A
GPO       CBT      ROUTINE                                                      GPO        1369     A
GPO                SX6    0                                                     GPO        1370     A
GPO                SA6    BSI         BSI = 0                                   GPO        1371     A
GPO                                                                              FCC0408      72     A
GPO       CBT1     SA1    BSI                                                   GPO        1372     A
GPO                SA2    O.BST                                                 GPO        1373     A
GPO                SA3    O.BIT                                                 GPO        1374     A
GPO                SX6    X1                                                    GPO        1375     A
GPO                SB4    X2                                                    GPO        1376     A
GPO                                                                              GPO        1377     A
GPO       CBT2     SX6    X6+2        BSI = BSI + 2                             GPO        1378     A
GPO                SA5    B4+X6       BAW = BST(BSI)                            GPO        1379     A
GPO                SB5    X3                                                    GPO        1380     A
GPO                ZR     X5,CBT0     IF BAW = 0    */ END OF *BST*             GPO        1381     A
GPO                LX5    -BA.FWAP                                              GPO        1382     A
GPO                SX7    X5-CBT                                                GPO        1383          I
          -FCC0675
GPO                MI     X7,CBT2     IF FWA[BAW] < CBT  */ DUMMY BLOCK         GPO        1384          I
          -FCC0675
GPO                LX5    BA.FWAP-BA.BIP  J = BI[BAW]                           GPO        1385          I
          -FCC0675
GPO                SA4    B5+X5       BIW = BIT(J)                              GPO        1386          I
          -FCC0675
GPO                SB6    X5-CBT                                                FCC0675     158     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                LX5    BA.FWAP-BA.BIP    J = BI[BAW]                          FCC0675     159    A
GPO                SA4    B5+X5       BIW = BIT(J)                               FCC0675     160    A
GPO                BX7    X4                                                     FCC0675     161           I
          -FCC1319
GPO                LX7    59-BI.RBP                                              FCC0675     162           I
          -FCC1319
GPO                MI     X7,CBT3     IF RB[BIW]    */ PROGRAM BLOCK             FCC0675     163           I
          -FCC1319
GPO                                                                              FCC0675     164           I
          -FCC1319
GPO                SA2    HBI                                                    FCC0675     165    A
GPO                MX0    -BI.OLNL                                               FCC0675     166    A
GPO                LX7    1+BI.RBP-BI.OLNP                                       FCC0675     167           I
          -FCC1319
GPO                BX7    X4                                                     FCC1319      71    A
GPO                LX7    -BI.OLNP                                               FCC1319      72    A
GPO                BX7    X0*X7                                                  FCC0675     168    A
GPO                BX7    X2+X7       OLN[BIT(J)] = HBI                          FCC0675     169    A
GPO                LX7    BI.OLNP                                                FCC0675     170    A
GPO                SA7    A4                                                     FCC0675     171    A
GPO                                                                              FCC0675     172    A
GPO       CBT3     MI     B6,CBT2     IF FWA[BAW] < CBT  */ DUMMY BLOCK          FCC0675     173    A
GPO                LX4    59-BI.CBP                                              GPO        1387    A
GPO                MI     X4,CBT2     IF CB[BAW]                                 GPO        1388    A
GPO                SA6    A1                                                     GPO        1389    A
GPO                CALL   RBS         RELEASE THE BLOCK                          GPO        1390           I
          -CCGNC
GPO                CALL   RBS#        RELEASE THE BLOCK                          CCGNC       622    A
GPO                EQ     CBT1                                                   GPO        1391    A
GPO       CIC      TITLE  CIC - CONVERT INTERVAL TO CODE                         GPO        1392    A
GPO      **        CIC - CONVERT INTERVAL TO CODE                                GPO        1393    A
GPO      *                                                                       GPO        1394    A
GPO      *         ENTRY  (IBA) = POINTER TO *IM* WORD OF INTERVAL               GPO        1395    A
GPO      *                (NN) = N. NODES IN INTERVAL                            GPO        1396    A
GPO      *                                                                       GPO        1397    A
GPO      *         EXIT   (IBA) = POINTS TO *IP* WORD OF NEXT INTERVAL           GPO        1398    A
GPO      *                INTERVAL CODED AND WRITTEN TO SCRATCH FILE.            GPO        1399    A
GPO                                                                              GPO        1400    A
GPO       CIC      ROUTINE                                                       GPO        1401    A
GPO                                                                              GPO        1402    A
GPO      *         SCAN INTERVAL LIST TO FORM AN *EBB*                           GPO        1403    A
GPO                                                                              GPO        1404    A
GPO       CIC1     SA1    IBA                                                    GPO        1405    A
GPO                SA3    O.BIT                                                  GPO        1406    A
GPO                SA2    A1+B1                                                  GPO        1407    A
GPO                SB3    X3                          (B3) = [O.BIT]             GPO        1408    A
GPO                SA5    X1          IP = [IBA]                                 GPO        1409    A
GPO                SB2    X2          N = NN                                     GPO        1410    A
GPO                BX2    X5          FIP = [IP]                                 GPO        1411    A
GPO                LX5    -IM.BIP     I = BI[IP]                                 GPO        1412    A
GPO                SA4    B3+X5                                                  GPO        1413    A
GPO                BX0    X4          FBIW = BIT(I)                              GPO        1414    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO              LX4    59-BI.RBP                                          GPO        1415     A
GPO              PL     X4,CIC3     IF ^RB[BIT(I)]  */ A HOLDING BLOCK     GPO        1416     A
GPO                                                                        GPO        1417     A
GPO      *       ADD SUCCESSORS THAT DO NOT HAVE AN INITIAL LABEL, ETC.    GPO        1418     A
GPO                                                                        GPO        1419     A
GPO       CIC2   SB2    B2-B1       N = N - 1                              GPO        1420     A
GPO              SA5    A5+B1       IP = IP + 1                            GPO        1421     A
GPO              ZR     B2,CIC4     IF N = 0      */ END OF LIST           GPO        1422     A
GPO              BX6    X2-X5                                              GPO        1423     A
GPO              LX6    59-IM.SCRP                                         GPO        1424     A
GPO              MI     X6,CIC4     IF SCR[IP] " SCR[FIP]  */ DIFF REGIONS GPO        1425     A
GPO              LX5    -IM.BIP     I = BI[IP]                             GPO        1426     A
GPO              SA4    B3+X5       BIW = BIT(I)                           GPO        1427     A
GPO              LX4    59-BI.ILP                                          GPO        1428          I
         -FCC0675
GPO              MI     X4,CIC4     IF IL[BIW]    */ INITIAL *LAB*         GPO        1429          I
         -FCC0675
GPO              LX4    BI.ILP-BI.RBP                                      GPO        1430          I
         -FCC0675
GPO              MI     X4,CIC2     IF RB[BIW]    */ NOT A HOLDING BLOCK   GPO        1431          I
         -FCC0675
GPO              LX4    59-BI.RBP                                          FCC0675     174     A
GPO              PL     X4,CIC3     IF ^RB[BIW]   */ A HOLDING BLOCK       FCC0675     175     A
GPO              LX4    BI.RBP-BI.ILP                                      FCC0675     176     A
GPO              PL     X4,CIC2     IF ^IL[BIW]                            FCC0675     177     A
GPO              EQ     CIC4                                               FCC0675     178     A
GPO                                                                        GPO        1432     A
GPO      *       TERMINATE LIST AFTER THIS NODE                            GPO        1433     A
GPO                                                                        GPO        1434     A
GPO       CIC3   SX6    A5+B1       IPL = IP + 1                           GPO        1435     A
GPO              EQ     CIC5                                               GPO        1436     A
GPO                                                                        GPO        1437     A
GPO       CIC4   SX6    A5+         IPL = IP      */ TERMINATE BEFORE      GPO        1438     A
GPO                                                                        GPO        1439     A
GPO       CIC5   LX0    59-BI.CBP                                          GPO        1440     A
GPO              IX6    X6-X1       K = IPL - IBA */ N. NODES IN *EBB*     GPO        1441     A
GPO              PL     X0,CIC7     IF ^CB[FBIW]  */ NOT YET CODED         GPO        1442     A
GPO                                                                        GPO        1443     A
GPO      *       ADVANCE POINTERS TO UNPROCESSED PART OF THE INTERVAL      GPO        1444     A
GPO                                                                        GPO        1445     A
GPO       CIC6   SA2    A1+B1                                              GPO        1446     A
GPO              IX7    X6+X1       IBA = IBA + K                          GPO        1447     A
GPO              SA7    A1                                                 GPO        1448     A
GPO              IX6    X2-X6       NN = NN - K                            GPO        1449     A
GPO              SA6    A2                                                 GPO        1450     A
GPO              NZ     X6,CIC1     IF NN " 0                              GPO        1451     A
GPO                                                                        GPO        1452     A
GPO              EQ     CIC                                                GPO        1453     A
GPO                                                                        GPO        1454     A
GPO      *       SET FP/LVL EXPANSION MODE, AND SETUP A *BST* FOR THIS *EBB*.  GPO     1455     A
GPO                                                                        GPO        1456     A
GPO       CIC7   SX0    B1                                                 GPO        1457     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO              LX2    -IM.SCRP                                          GPO        1458    A
GPO              BX3    X0*X2                                             GPO        1459    A
GPO              SX7    X3+3        EMF = 3+SCR[FIP]  */ FP/LVL EXPANSION FLG  GPO   1460    A
GPO              SA7    EMF                                               GPO        1461    A
GPO              BX5    X6                                                GPO        1462    A
GPO              LX2    B1,X6                                             GPO        1463    A
GPO              SA6    CICA                                              GPO        1464    A
GPO              MX0    -IM.BIL                                           GPO        1465    A
GPO              ALLOC  BST,X2+3    ALLOC( BST , 2*K+3 )                  GPO        1466    A
GPO              SA1    IBA                                               GPO        1467    A
GPO              SB2    X5          N = K                                 GPO        1468    A
GPO              MX7    0                                                 GPO        1469    A
GPO              SA5    X1          IP = [IBA]                            GPO        1470    A
GPO              SA7    X2          BST(0) = 0                            GPO        1471    A
GPO              LX0    IM.BIP                                            GPO        1472    A
GPO              SA7    X2+B1       BST(1) = 0;  I = 2                    GPO        1473    A
GPO                                                                       GPO        1474    A
GPO       CIC8   BX6    -X0*X5                                            GPO        1475    A
GPO              LX6    BA.BIP-IM.BIP                                     GPO        1476    A
GPO              SA6    A7+B1       BST(I) = BAW(0,BI[IP],0,0)            GPO        1477    A
GPO              BX7    X5                                                GPO        1478    A
GPO              SA7    A6+B1       BST(I+1) = [IP];  I = I + 2           GPO        1479    A
GPO              SB2    B2-B1       N = N - 1                             GPO        1480    A
GPO              SA5    A5+B1       IP = IP + 1                           GPO        1481    A
GPO              NZ     B2,CIC8     IF N " 0                              GPO        1482    A
GPO                                                                       GPO        1483    A
GPO              MX7    0                                                 GPO        1484    A
GPO              SA7    A7+B1       BST(I) = 0    */ *BST* TERMINATOR     GPO        1485    A
GPO              SA7    MAXW        MAXW = 0                              GPO        1486    A
GPO              SA7    N.EBB       N.EBB = 0                             GPO        1487    A
GPO                                                                       GPO        1488    A
GPO              CLCM   CHB         COMBINE HOLDING BLOCKS                GPO        1489    A
GPO                                                                       GPO        1490    A
GPO              CLCM   RDD         REMOVE DEAD DEFS                      GPO        1491    A
GPO                                                                       GPO        1492    A
GPO              CALL   CBB         COMBINE BASIC BLOCKS                  GPO        1493    A
GPO                                                                       GPO        1494    A
GPO              CLCM   CXB         CODE EXTENDED BLOCKS                  GPO        1495         I
          -CCGNC
GPO              CLCM   CXB#        CODE EXTENDED BLOCKS                  CCGNC       623    A
GPO                                                                       GPO        1496    A
GPO              RJ     CBT         CLEAR *BST*                           GPO        1497    A
GPO              SA2    CICA                                              GPO        1498    A
GPO              SA1    IBA                                               GPO        1499    A
GPO              BX6    X2                                                GPO        1500    A
GPO              EQ     CIC6                                              GPO        1501    A
GPO                                                                       GPO        1502    A
GPO       CICA   BSS    1                                                 GPO        1503    A
GPO       CBB    TITLE  CBB - COMBINE BASIC BLOCKS TO FORM EXTENDED BLOCK GPO        1504    A
GPO      **      CBB - COMBINE BASIC BLOCKS TO FORM AN EXTENDED BLOCK.    GPO        1505    A
GPO      *                                                                GPO        1506    A
GPO      *       ENTRY  (EMF) = FP/LEVEL EXPANSION MODE                   GPO        1507    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO       *                                                                GPO        1508    A
GPO       *          EXIT   EXTENDED BLOCK WRITTEN BACK TO *BLK*           GPO        1509    A
GPO       *                 MAXW, N.EBB UPDATED.                           GPO        1510    A
GPO                                                                        GPO        1511    A
GPO        CBB       ROUTINE                                               GPO        1512    A
GPO                  SX6    2           BSIL = 2                           GPO        1513    A
GPO                  SA6    BSIL                                           GPO        1514    A
GPO                                                                        GPO        1515    A
GPO        CBB1      SA1    BSIL        I = BSIL                           GPO        1516    A
GPO                  SA2    O.BST                                          GPO        1517    A
GPO                  SA3    O.BIT                                          GPO        1518    A
GPO                  SA4    M.EBL                                          GPO        1519    A
GPO                  SB3    X2                       (B3) = [O.BST]        GPO        1520    A
GPO                  BX6    X1          BSI = BSIL                         GPO        1521    A
GPO                  SA6    BSI                                            GPO        1522    A
GPO                  SB4    X3                       (B4) = [O.BIT]        GPO        1523    A
GPO                  SB7    X4                       (B7) = M.EBL          GPO        1524    A
GPO                  SA5    B3+X1       BAW = BST(I)                       GPO        1525    A
GPO                  SA0    X1                                             GPO        1526    A
GPO                  ZR     X5,CBB      IF BAW = 0    */ END OF *BST*      GPO        1527    A
GPO                                                                        GPO        1528    A
GPO                  LX5    -BA.LENP                                       GPO        1529    A
GPO                  SB6    X5          TLEN = LEN[BAW]  */ TOTAL LENGTH OF EBB  GPO  1530    A
GPO                  LX5    BA.LENP-BA.BIP                                 GPO        1531    A
GPO                  SA4    B4+X5       BIW = BIT(BI[BAW])                 GPO        1532    A
GPO                  LX4    59-BI.RBP                                      GPO        1533    A
GPO                  PL     X4,CBB3     IF ^RB[BIW]   */ HOLDING BLOCK     GPO        1534    A
GPO                  SB2    2                                              GPO        1535    A
GPO                  SX7    X5+         LBN = BI[BAW]                      GPO        1536    A
GPO                                                                        GPO        1537    A
GPO       *          ADD SUCCESSORS OF THIS BLOCK UNTIL *IL*, SPACE OVERFLOW, ETC.  GPO  1538  A
GPO                                                                        GPO        1539    A
GPO        CBB2      SX1    X1+B2       I = I + 2                          GPO        1540    A
GPO                  LX4    BI.RBP-BI.FJP                                  GPO        1541    A
GPO                  GT     B6,B7,CBB3A IF TLEN > M.EBL  */ SPACE OVERFLOW GPO        1542    A
GPO                  MI     X4,CBB4     IF FJ[BIW]                         GPO        1543    A
GPO                  SX7    X7+B2       LBN = LBN + 2                      GPO        1544    A
GPO                  SA5    B3+X1       BAW = BST(I)                       GPO        1545    A
GPO                  ZR     X5,CBB4     IF BAW = 0    */ END OF *BST*      GPO        1546    A
GPO                  LX5    -BA.BIP     BN = BI[BAW]                       GPO        1547    A
GPO                  SA4    B4+X5       BIW = BIT(BN)                      GPO        1548    A
GPO                  LX4    59-BI.ILP                                      GPO        1549    A
GPO                  MI     X4,CBB4     IF IL[BIW]                         GPO        1550    A
GPO                  BX6    X5-X7                                          GPO        1551    A
GPO                  SB5    X6                                             GPO        1552    A
GPO                  LX5    BA.BIP-BA.LENP                                 GPO        1553    A
GPO                  SB6    B6+X5       TLEN = TLEN + LEN[BAW]             GPO        1554    A
GPO                  LX4    BI.ILP-BI.RBP                                  GPO        1555    A
GPO                  PL     X4,CBB3     IF ^RB[BIW]                        GPO        1556          I
          -FCCA509X
GPO                  PL     X4,CBB3B    IF ^RB[BIW]                        FCCA509X      8    A
GPO                  ZR     B5,CBB2     IF BN[BAW] = LBN  */ BLOCKS CONTIGIOUS  GPO  1557    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                EQ     CBB4                                                    GPO          1558     A
GPO                                                                               GPO          1559     A
GPO        CBB3    SX6    X1+2        LIM = I + 2                                 GPO          1560     A
GPO                EQ     CBB5                                                    GPO          1561     A
GPO                                                                               GPO          1562     A
GPO        CBB3A   SB7    B7+300B                                                 GPO          1563     A
GPO                LE     B6,B7,CBB4  IF TLEN @ M.EBL + 300B                      GPO          1564     A
GPO                                                                               GPO          1565     A
GPO                SX2    A0+B2                                                   GPO          1566     A
GPO                IX7    X1-X2                                                   GPO          1567     A
GPO                ZR     X7,CBB4     IF I = BSIL + 2  */ ONLY 1 BIG BLOCK        GPO          1568     A
GPO                SX1    X1-2        I = I - 2                                   GPO          1569     A
GPO                EQ     CBB4                                                    FCCA509X        9     A
GPO                                                                               FCCA509X       10     A
GPO        *       ENCOUNTERED *HB* IN SEQ - CHECK IF THE LOOP HEADER BLK         FCCA509X       11     A
GPO        *       IS A DIRECT SUCCESSOR OF THE PREV. BLK IN THE BST.             FCCA509X       12     A
GPO        *       IF SO INCLUDE THE *HB* IN THE *EBB*.                           FCCA509X       13     A
GPO                                                                               FCCA509X       14     A
GPO        CBB3B   LX4    BI.RBP-BI.PBNP+1       HDRBLK = PBN[BIW]                FCCA509X       15     A
GPO                SB5    X1-2                                                    FCCA509X       16     A
GPO                SA2    B3+B5                                                   FCCA509X       17     A
GPO                LX2    -BA.BIP     PBLK = BI[BST(I-2)]                         FCCA509X       18     A
GPO                IX7    X4-X2                                                   FCCA509X       19     A
GPO                SB5    X7                                                      FCCA509X       20     A
GPO                NE     B5,B2,CBB4  IF PBLK+2 " HDRBLK                          FCCA509X       21     A
GPO                SX1    X1+B2                                                   FCCA509X       22     A
GPO                                                                               FCCA509X       23     A
GPO                                                                               GPO          1570     A
GPO        CBB4    SX6    X1+         LIM = I                                     GPO          1571     A
GPO                                                                               GPO          1572     A
GPO        CBB5    SA6    A1+B1       BSIL(2) = LIM                               GPO          1573     A
GPO                SX7    4                                                       GPO          1574     A
GPO                SB5    B0          J = 0                                       GPO          1575     A
GPO                SA7    L.TXT       L.TXT = 4                                   GPO          1576     A
GPO                SB6    A0          I = BSIL                                    GPO          1577     A
GPO                SB7    X6                                                      GPO          1578     A
GPO                                                                               GPO          1579     A
GPO        *       COUNT NUMBER OF NON-EMPTY BLOCKS IN *EBB*                      GPO          1580     A
GPO                                                                               GPO          1581     A
GPO        CBB6    SA5    B3+B6       BAW = BST(I)                                GPO          1582     A
GPO                SB6    B6+2        I = I + 2                                   GPO          1583     A
GPO                LX5    -BA.LENP                                                GPO          1584     A
GPO                SB2    X5-8                                                    GPO          1585     A
GPO                LE     B2,CBB7     IF LEN[BAW] @ 8                             GPO          1586     A
GPO                                                                               GPO          1587     A
GPO                NZ     B5,CBB10    IF J " 0      */ MORE THAN 1 BLOCK IN *EBB*  GPO          1588          I
           -FCC0643
GPO                NZ     B5,CBB8     IF J " 0      */ MORE THAN 1 NON-EMPTY BLOCK FCC0643        9     A
GPO                SB5    B6-2        J = I - 2     */ BST INDX OF FIRST           GPO          1589     A
GPO                                                                               GPO          1590     A
GPO        CBB7    LT     B6,B7,CBB6  IF I < BSIL(2)                              GPO          1591     A
GPO                                                                               GPO          1592     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                 ZR     B5,CBB15    IF J = 0      */ NO NON-EMPTY BLOCKS                GPO         1593     A
GPO                                                                                        GPO         1594     A
GPO                 SX6    B5+         BSI = J                                             GPO         1595     A
GPO                 SA6    BSIL        BSIL = J                                            GPO         1596     A
GPO                 SA6    BSI                                                             GPO         1597     A
GPO                 CALL   RTB         READ BLOCK IN                                       GPO         1598          I
          -FCC0643
GPO                 SA5    EMF                                                             GPO         1599          I
          -FCC0643
GPO                 MX6    1                                                               GPO         1600          I
          -FCC0643
GPO                 CALL   ESR         ESR( EMF , 1S59 )  */ EXPAND SPECIAL REFS           GPO         1601          I
          -FCC0643
GPO                 SA1    L.MLT                                                           GPO         1602          I
          -FCC0643
GPO                 ZR     X1,CBB8     IF L.MLT = 0  */ NO SPECIAL REFS                    GPO         1603          I
          -FCC0643
GPO                 MX6    0                                                               GPO         1604          I
          -FCC0643
GPO                 CALL   MPB         MERGE MODS AND MOVE BLOCK TO *TXT*                  GPO         1605          I
          -FCC0643
GPO                 EQ     CBB12                                                           GPO         1606          I
          -FCC0643
GPO                                                                                        GPO         1607          I
          -FCC0643
GPO       *         MOVE INSTRUCTIONS TO *TXT*, CLEAR LINK WORDS                            GPO         1608          I
          -FCC0643
GPO                                                                                        GPO         1609          I
          -FCC0643
GPO        CBB8     CALL   MBT         MOVE BLOCK TO *TXT*                                 GPO         1610          I
          -FCC0643
GPO                 EQ     CBB13                                                           GPO         1611          I
          -FCC0643
GPO                                                                                        GPO         1612          I
          -FCC0643
GPO       *         FORM AN EXTENDED BLOCK WITH THE SPECIAL MEM REFS EXPANDED               GPO         1613          I
          -FCC0643
GPO                                                                                        GPO         1614          I
          -FCC0643
GPO        CBB10    SA5    O.BST                                                           GPO         1615          I
          -FCC0643
GPO                 EQ     CBB10                                                           FCC0643       10     A
GPO                                                                                        FCC0643       11     A
GPO       *         FORM AN EXTENDED BASIC BLOCK ( *EBB* )                                  FCC0643       12     A
GPO                                                                                        FCC0643       13     A
GPO        CBB8     SA5    O.BST                                                           FCC0643       14     A
GPO                 SA4    BSI                                                             GPO         1616     A
GPO                 SB5    X5                                                              GPO         1617     A
GPO                 SA1    B5+X4       BAW = BST(BSI)                                      GPO         1618     A
GPO                 LX1    -BA.LENP                                                        GPO         1619     A
GPO                 SB2    X1-8                                                            GPO         1620     A
GPO                 LE     B2,CBB11    IF LEN[BAW] @ 8  */ EMPTY BLOCK                     GPO         1621          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

          -FCC0643
GPO                LE     B2,CBB9     IF LEN[BAW] @ 8  */ EMPTY BLOCK                FCC0643      15     A
GPO                                                                                  GPO         1622     A
GPO                CALL   RTB         READ BLOCK IN                                  GPO         1623          I
          -CCGNC
GPO                CALL   RTB#        READ BLOCK IN                                  CCGNC        624     A
GPO                SA2    O.SEQ       T = [O.SEQ]                                    GPO         1624     A
GPO                SB2    OC.EOS                                                     GPO         1625     A
GPO                SA4    =XF.RDT+B2                                                 GPO         1626     A
GPO                SA5    EMF                                                        GPO         1627          I
          -FCC0643
GPO                SA3    X2                                                         GPO         1628     A
GPO                PX7    B2,X3       OC[T] = OC.EOQ                                 GPO         1629     A
GPO                SA7    A3                                                         GPO         1630     A
GPO                MX6    0                                                          GPO         1631     A
GPO                BX7    X4                                                         GPO         1632     A
GPO                SA7    A7+2        [T+2] = RDT(OC.EOS)                            GPO         1633     A
GPO                CALL   ESR  ( EMF , 0 )  EXPAND SPECIAL MEM REFS                  GPO         1634          I
          -FCC0643
GPO                CALL   MPB ( 0 )   MERGE BLOCKS                                  FCC0643      16     A
GPO                                                                                  GPO         1635     A
GPO                CALL   RBS         RELEASE THE BLOCK                              GPO         1636          I
          -CCGNC
GPO                CALL   RBS#        RELEASE THE BLOCK                              CCGNC        625     A
GPO       CBB11    SA1    BSI                                                        GPO         1637          I
          -FCC0643
GPO       CBB9     SA1    BSI                                                        FCC0643      17     A
GPO                SA2    BSIL+1                                                     GPO         1638     A
GPO                SX6    X1+2        BSI = BSI + 2                                  GPO         1639     A
GPO                IX7    X6-X2                                                      GPO         1640     A
GPO                SA6    A1                                                         GPO         1641     A
GPO                MI     X7,CBB10    IF BSI < BSIL(2)                               GPO         1642          I
          -FCC0643
GPO                MI     X7,CBB8     IF BSI < BSIL(2)                               FCC0643      18     A
GPO                                                                                  GPO         1643     A
GPO                SA1    O.TXT                                                      GPO         1644     A
GPO                SA2    X1          T = [O.TXT]                                    GPO         1645     A
GPO                PX7    B1,X2       OC[T] = OC.BOS                                 GPO         1646     A
GPO                SA7    A2+                                                        GPO         1647     A
GPO                                                                                  GPO         1648          I
          -FCC0643
GPO       CBB12    MX6    0                                                          GPO         1649          I
          -FCC0643
GPO                CALL   SQZB        SQUEEZE THE EXTENDED BLOCK                     GPO         1650          I
          -FCC0643
GPO                SA1    BSIL                                                       FCC0643      19     A
GPO                MX6    0                                                          FCC0643      20     A
GPO                BX7    X1          BSI = BSIL(1)                                  FCC0643      21     A
GPO                SA7    BSI                                                        FCC0643      22     A
GPO                CALL   SQZB        SQUEEZE THE EXTENDED BLOCK                     FCC0643      23          I
          -CCGNC
GPO                CALL   SQZB#       SQUEEZE THE EXTENDED BLOCK                     CCGNC        626     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                SA1    EMF                                                  FCC0643       24     A
GPO                SA2    =XESRA-1+X1                                          FCC0643       25     A
GPO                SA3    X2                                                   FCC0643       26     A
GPO                ZR     X3,CBB13    IF [ESRA(EMF)] = 0  */ NO FP OR L2 IN PROG FCC0643     27     A
GPO                CALL   WTB         MOVE EBB FROM *TXT* TO *BLK*             FCC0643       28     A
GPO                                                                            FCC0643       29     A
GPO       *        EXPAND SPECIAL MEMORY REFS AND MERGE/SQZ BLOCK              FCC0643       30     A
GPO                                                                            FCC0643       31     A
GPO        CBB10   CALL   RTB         READ BLOCK IN                            FCC0643       32          I
          -CCGNC
GPO        CBB10   CALL   RTB#        READ BLOCK IN                            CCGNC        627     A
GPO                IFEQ   HC.ID,2,1         FTN ONLY                           CCG29          7     A
GPO                CALL   SLC         SUBSTITUTE LOCAL COPIES                  CCG23          1     A
GPO                SA5    EMF                                                  FCC0643       33     A
GPO                MX6    1                                                    FCC0643       34     A
GPO                CALL   ESR         ESR( EMF , 1S59 )  */ EXPAND SPECIAL REFS FCC0643       35          I
          -CCGNC
GPO                CALL   ESR#        ESR( EMF , 1S59 )  */ EXPAND SPECIAL REFS CCGNC        628     A
GPO                SA1    L.MLT                                                FCC0643       36     A
GPO                ZR     X1,CBB11    IF L.MLT = 0  */ NO MODS TO THIS BLOCK   FCC0643       37     A
GPO                SX6    B1                                                   FCC0643       38     A
GPO                CALL   MPB ( 1 )   MERGE MODS AND SQUEEZE THE BLOCK         FCC0643       39     A
GPO                EQ     CBB13                                                FCC0643       40     A
GPO                                                                            FCC0643       41     A
GPO        CBB11   CALL   MBT         MOVE BLOCK TO *TXT*                      FCC0643       42     A
GPO                                                                            GPO          1651     A
GPO        CBB13   SX6    B1                                                   GPO          1652     A
GPO                CALL   RIO         REORDER THE INSTRUCTIONS                 GPO          1653          I
          -CCGNC
GPO                CALL   RIO#        REORDER THE INSTRUCTIONS                 CCGNC        629     A
GPO                SA2    MAXW                                                 GPO          1654     A
GPO                SA1    BSIL                                                 GPO          1655          I
          -FCC0643
GPO                SA4    N.EBB                                                FCC0643       43     A
GPO                IX3    X2-X6                                                GPO          1656     A
GPO                PL     X3,CBB14    MAXW = MAX( MAXW , SIO.MAX )             GPO          1657     A
GPO                SA6    A2                                                   GPO          1658     A
GPO        CBB14   SA3    N.EBB                                                GPO          1659          I
          -FCC0643
GPO                BX7    X1                                                   GPO          1660          I
          -FCC0643
GPO                SX6    X3+B1       N.EBB = N.EBB + 1                        GPO          1661          I
          -FCC0643
GPO                SA7    BSI         BSI = BSIL(1)                            GPO          1662          I
          -FCC0643
GPO                SA6    A3+                                                  GPO          1663          I
          -FCC0643
GPO        CBB14   SX6    X4+B1       N.EBB = N.EBB + 1                        FCC0643       44     A
GPO                SA6    A4                                                   FCC0643       45     A
GPO                CALL   WTB         WRITE BLOCK BACK TO *BLK*                GPO          1664     A
GPO                                                                            GPO          1665     A
GPO        CBB15   SA5    BSIL+1                                               GPO          1666     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                BX7    X5          BSIL(1) = BSIL(2)                          GPO        1667    A
GPO                SA7    A5-B1                                                  GPO        1668    A
GPO                EQ     CBB1                                                   GPO        1669    A
GPO                                                                              GPO        1670    A
GPO       BSIL     BSS    2           *BST* LIMITS                               GPO        1671    A
GPO       ATT      EJECT                                                          FCC0514      31    A
GPO      **        ATT - ADJUST *TET* TABLE FOR *GRA* AND *BDT*.  CLEAR *REG* FIE FCC0514      32    A
GPO      *         FIELD, UPDATE INFO OF EQUIVALENCED ENTRIES.                    FCC0514      33    A
GPO                                                                              FCC0514      34    A
GPO       ATT      ROUTINE                                                        FCC0514      35    A
GPO                SA2    ITL                                                    FCC0514      36    A
GPO                SA3    N.GT                                                   FCC0514      37    A
GPO                SA1    O.TET                                                  FCC0514      38    A
GPO                SB2    X2          I = ITL                                    FCC0514      39    A
GPO                SB3    X3          L = N.GT                                   FCC0514      40    A
GPO                EQ     B2,B3,ATT3  IF ITL = N.GT */ NO TET"S CREATED          FCC0514      41    A
GPO                                                                              FCC0514      42    A
GPO      *         CLEAR *REG* FIELD OF *TET*"S CREATED IN THIS LOOP FOR *GRA*.   FCC0514      43    A
GPO                                                                              FCC0514      44    A
GPO                MX0    -T.REGL                                                FCC0514      45    A
GPO                LX0    T.REGP                                                 FCC0514      46    A
GPO                MX4    -1                                                     CCGA049      13    A
GPO                LX4    T.CALP                      */ SEE STS IN SQZ          CCGA049      14    A
GPO                BX0    X0*X4                       */ UNLOCK CA FIELD         CCGA049      15    A
GPO       ATT2     SA2    X1+B2                                                  FCC0514      47    A
GPO                BX6    X0*X2       REG[TET(I)] = 0                            FCC0514      48    A
GPO                SB2    B2+B1       I = I + 1                                  FCC0514      49    A
GPO                SA6    A2                                                     FCC0514      50    A
GPO                LT     B2,B3,ATT2  IF I < L                                   FCC0514      51    A
GPO                                                                              FCC0514      52    A
GPO       ATT3     SA4    L.TET                                                  FCC0514      53    A
GPO                SB5    X1                                                     FCC0514      54    A
GPO                SB4    X4          I = L.TET                                  FCC0514      55    A
GPO                EQ     B4,B3,ATT   IF L.TET = N.GT  */ NO TST SQUEEZING       FCC0514      56    A
GPO                                                                              FCC0514      57    A
GPO      *         MOVE *IP* FIELDS OF THE EQUIVED ENTRIES TO THEIR BASE MEMBERS. FCC0514      58          I
         -CCG0127
GPO      *         MOVE THE *CA* FIELDS OF THE SQUEEZED ENTRIES TO TET(CA)        CCG0127       9    A
GPO                                                                              FCC0514      59    A
GPO                MX0    -T.IPFL                                                FCC0514      60    A
GPO                SX6    B3          L.TET = N.GT                               FCC0514      61    A
GPO                LX0    T.IPFP                                                 FCC0514      62    A
GPO                MX1    1+T.CAL                                                FCC0514      63    A
GPO                SA6    A4                                                     FCC0514      64    A
GPO                LX1    T.CAL                                                  FCC0514      65    A
GPO                                                                              FCC0514      66    A
GPO       ATT4     SB4    B4-B1       I = I - 1                                  FCC0514      67    A
GPO                SA2    B5+B4       TI = TET(I)                                FCC0514      68    A
GPO                BX7    X1*X2       C = CA[TI]                                 FCC0514      69    A
GPO                SA3    B5+X2                                                  FCC0514      70    A
GPO                BX4    -X0*X3      IPI = IPF[TET(C)]                          FCC0514      71    A
GPO                LX3    59-T.EQVP                                              CCG0127      10    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                MI     X3,ATT5     IF EQV[TET(C)]  */ CA .NE. C                    CCG0127      11     A
GPO                IX6    X4+X7       TET(C) = TETW(INV[TI],IPI,0,C)                  FCC0514      72     A
GPO                SA6    A3+                                                         FCC0514      73     A
GPO                GT     B4,B3,ATT4  IF I > L                                        FCC0514      74          I
          -CCG0127
GPO        ATT5    GT     B4,B3,ATT4  IF I > L                                        CCG0127      12     A
GPO                                                                                   FCC0514      75     A
GPO                EQ     ATT                                                         FCC0514      76     A
GPO        SLC     SPACE  3,14                                                        CCG23         2     A
GPO        .FPLC   IFEQ   HC.ID,2                                                     CCG29         8     A
GPO                                                                                   CCG29         9     A
GPO       **       SLC - SUBSTITUTE LOCAL COPIES OF FORMAL PARAMETERS                 CCG23         3     A
GPO       *                                                                           CCG23         4     A
GPO       *        ENTRY  CURRENT SEQUENCE IN *BLK*, ESR NOT YET CALLED               CCG23         5     A
GPO                                                                                   CCG23         6     A
GPO        SLC     ROUTINE                                                            CCG23         7     A
GPO                SA1    HO$OPT                                                      CCG23         8     A
GPO                PL     X1,SLC      IF HO$OPT " 2                                   CCG23         9     A
GPO                SA2    =XL$FPI                                                     CCG23        10     A
GPO                ZR     X2,SLC      IF L$FPI = 0  */ NO F.P. S                      CCG23        11     A
GPO                SA4    O.SEQ                                                       CCG23        12     A
GPO                SA3    HBI                                                         CCG23        13     A
GPO                SX0    RN.MASK                                                     CCG23        14     A
GPO                SB4    4                                                           CCG23        15     A
GPO                SB5    OC.STT                                                      CCG23        16     A
GPO                SB6    X3                (B6) = HBI  */ =0 IF OUTSIDE OF LPS       CCG23        17     A
GPO                SA5    X4          R1 = O.TXT                                      CCG23        18     A
GPO                EQ     SLC3                                                        CCG23        19     A
GPO                                                                                   CCG23        20     A
GPO       *        IF LOOP MODE THEN ONLY CALL *MPR* FOR *LCM* F.P. REFS              CCG23        21     A
GPO                                                                                   CCG23        22     A
GPO        SLC1    EQ     B2,B5,SLC3  IF OC[R1] = OC.STT                              CCG23        23     A
GPO                ZR     B6,SLC2     IF HBI = 0    */ OUTSIDE OF ALL LOOPS           CCG23        24     A
GPO                LX4    D.FPP-D.L2P                                                 CCG23        25     A
GPO                PL     X4,SLC3     IF ^L2[DI]                                      CCG23        26     A
GPO                LX4    D.L2P-D.PSP                                                 CCGA042       6     A
GPO                MI     X4,SLC3     IF PS[DI]                                       CCGA042       7     A
GPO                                                                                   CCG23        27     A
GPO        SLC2    SX7    B6                                                          CCG23        28     A
GPO                CALL   MPR#        CALL MPR( R1 , HBI )                            CCG23        29     A
GPO                                                                                   CCG23        30     A
GPO        SLC3    SA5    A5+B4       R1 = R1 + 4                                     CCG23        31     A
GPO                SA4    A5+2        DI = R1 + 2                                     CCG23        32     A
GPO                UX6    B2,X5                                                       CCG23        33     A
GPO                LX4    59-D.FPP                                                    CCG23        34     A
GPO                MI     X4,SLC1     IF FP[DI]                                       CCG23        35     A
GPO                NZ     B2,SLC3     IF OC[R1] " 0   */ IF NOT END OF *TXT*          CCG23        36     A
GPO                                                                                   CCG23        37     A
GPO                EQ     SLC                                                         CCG23        38     A
GPO        MPR     SPACE  3,14                                                        CCG23        39     A
GPO       **       MPR - MODIFY PARAMETER REFERENCE                                   CCG23        40     A
GPO       *                                                                           CCG23        41     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO      *                CHANGE LD/ST TO A F.P. TO A REFERENCE TO LC.+CA          CCG23    42    A
GPO      *          DONE FOR SCALAR VARIABLES THAT ARE NOT TYPE CHAR               CCG23    43    A
GPO      *                                                                         CCG23    44    A
GPO      *          ENTRY  (A5,X5) = R1, [R1]                                      CCG23    45    A
GPO      *                (X7) = FLAG, =0 IF OUTSIDE OF ALL LOOPS                  CCG23    46    A
GPO                                                                                CCG23    47    A
GPO       MPR       ROUTINE                                                        CCG23    48    A
GPO                 SA4    A5+B1       R2 = R1 + 1                                 CCG23    49    A
GPO                 SB3    X4                                                      CCG23    50    A
GPO                 LX4    -IH.RFP                                                 CCG23    51    A
GPO                 BX6    "RN"X4                                                  CCG23    52    A
GPO                 NZ     X6,MPR      IF RF[R2] " 0  */ INDEXED REFERENCE         CCG23    53    A
GPO                 SA1    O.SYM                                                   CCG23    54    A
GPO                 LX4    IH.RFP                                                  CCG23    55    A
GPO                 SB2    B3+B3                                                   CCG23    56    A
GPO                 SB2    B2+X1                                                   CCG23    57    A
GPO                 SB3    B3+B1                                                   CCG23    58    A
GPO                 SA1    B2+B3       WORDB = SYM(3*IH[R2]+1)                     CCG23    59    A
GPO                 MX2    -WB.MODEL                                               CCG23    60    A
GPO                 LX1    -WB.MODEP                                               CCG23    61    A
GPO                 BX3    -X2*X1                                                  CCG23    62    A
GPO                 SX6    X3-M.CHAR                                               CCG23    63    A
GPO                 ZR     X6,MPR      IF MODE[WORDB] = M.CHAR                     CCG23    64    A
GPO                 SX3    X3-M.DBL+8                                              CCG23    65    A
GPO                 AX3    3                                                       CCG23    66    A
GPO                 SB3    X3+B1       SDPF = (MODE-M.DBL+8)/8 + 1                 CCG23    67    A
GPO                 LX1    59-WB.ARYP+WB.MODEP                                     CCG23    68    A
GPO                 MI     X1,MPR      IF ARY[WORDB]   */ IF NOT A SCALAR          CCG23    69    A
GPO                 LX1    WB.ARYP-WB.LOCFP                                        CCG23    70    A
GPO                 MX2    -WB.FPOL                                                CCG23    71    A
GPO                 MI     X1,MPR      IF LOCF[WORDB]  */ REFED IN LOCF ARG LIST   CCG23    72    A
GPO                 SA3    =XO$FPI                                                 CCG23    73    A
GPO                 LX1    1+WB.LOCFP-WB.FPOP                                      CCG23    74    A
GPO                 BX6    -X2*X1                                                  CCG23    75    A
GPO                 IX6    X3+X6                                                   CCG23    76    A
GPO                 SA2    X6-1        FPW = FPI(FPO[WORDB])                       CCG23    77    A
GPO                 BX3    X2                                                      CCG23    78    A
GPO                 LX3    59-FP.LCP                                               CCG23    79    A
GPO                 MI     X3,MPR4     IF LC[FPW]    */ IF A LOCAL COPY EXISTS     CCG23    80    A
GPO                                                                                CCG23    81    A
GPO                 NZ     X7,MPR1     IF FLAG " 0   */ INSIDE A LOOP              CCG23    82    A
GPO                 SA3    A1+B1       WORDC = WORDB + 1                           CCG23    83    A
GPO                 SX7    X3-3                                                    CCG23    84    A
GPO                 MI     X7,MPR      IF N.REFS[WORDC] < 3                        CCG23    85    A
GPO                                                                                CCG23    86    A
GPO       MPR1      SA3    =XN$SEP                                                 CCG23    87    A
GPO                 SKIP   1           REMOVE WHEN FTN 5 IS ABLE TO HANDLE DEF CASE CCG23    88    A
GPO                 ZR     X3,MPR2     IF N$SEP = 0  */ NO SECONDARY ENTRY PTS     CCG23    89    A
GPO                 LX1    WB.FPOP-1-WB.DEFP                                       CCG23    90    A
GPO                 MI     X1,MPR      IF DEF[WORDB] */ CANT HANDLE DEF AND ENT PTS CCG23   91    A
GPO                                                                                CCG23    92    A
GPO      *          ASSIGN A NEW LC.                                               CCG23    93    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                                                                                  CCG23        94     A
GPO       MPR2     SA3    =XN$LC                                                    CCG23        95     A
GPO                SX7    B1                                                        CCG23        96     A
GPO                LX7    FP.LCP                                                    CCG23        97     A
GPO                LX3    FP.CAP       CA[FPW] = N$LC                               CCG23        98     A
GPO                BX7    X3+X7        LC[FPW] = 1                                  CCG23        99     A
GPO                LX3    -FP.CAP                                                   CCG23       100     A
GPO                BX2    X2+X7                                                     CCG23       101     A
GPO                BX6    X2                                                        CCG23       102     A
GPO                SA6    A2                                                        CCG23       103     A
GPO                SX7    X3+B3        N$LC = N$LC + SDPF                           CCG23       104     A
GPO                SA7    A3                                                        CCG23       105     A
GPO                                                                                 CCG23       106     A
GPO      *         MODIFY THE MEMORY REFERENCE                                      CCG23       107     A
GPO                                                                                 CCG23       108     A
GPO       MPR4     LX2    -FP.CAP                                                   CCG23       109     A
GPO                SX2    X2                                                        CCG23       110           I
         -CCGA043
GPO                MX3    -FP.CAL                                                   CCGA043       6     A
GPO                BX2    -X3*X2       EXTRACT FP.CA                                CCGA043       7     A
GPO                LX2    IH.CAP                                                    CCG23       111     A
GPO                IX4    X2+X4        CA[R2] = CA[R2] + CA[FPW]                    CCG23       112     A
GPO                SA3    =XS$LC                                                    CCG23       113     A
GPO                MX1    -IH.IHL                                                   CCG23       114     A
GPO                BX4    X1*X4                                                     CCG23       115     A
GPO                BX6    X3+X4        IH[R2] = S$LC                                CCG23       116     A
GPO                SA6    A4                                                        CCG23       117     A
GPO                SA2    A4+B1        DI = R2 + 1                                  CCG23       118     A
GPO                SA3    MPRA                                                      CCG23       119     A
GPO                BX6    -X3*X2       (FP,L2)[DI] = 0                              CCG23       120           I
         -CCGA088
GPO                BX6    -X3*X2       (FP,L2,XMT)[DI] = 0                          CCGA088       6     A
GPO                SA6    A2                                                        CCG23       121     A
GPO                EQ     MPR                                                       CCG23       122     A
GPO                                                                                 CCG23       123     A
GPO       MPRA     BFMW   D,(FP,L2)                                                 CCG23       124           I
         -CCGA088
GPO       MPRA     BFMW   D,(L2,XMT,FP)                                             CCGA088       7     A
GPO                                                                                 CCG29        10     A
GPO       .FPLC    ELSE                                                             CCG30         2     A
GPO       MPR      ROUTINE                                                          CCG30         3     A
GPO                EQ     MPR                                                       CCG30         4     A
GPO                                                                                 CCG30         5     A
GPO       .FPLC    ENDIF                                                            CCG29        11     A
GPO       CRB      SPACE  3,18         CRB                                          GPO        1672     A
GPO      **        CRB - CLEAR REGION BITS IN *UDT*                                 GPO        1673     A
GPO      *                                                                          GPO        1674     A
GPO      *         ENTRY  (X5) = 0 IF N.BLK IN REGION = 1                           GPO        1675     A
GPO                                                                                 GPO        1676     A
GPO       CRB2     SA1    O.UDT                                                     GPO        1677     A
GPO                SA2    N.VAR                                                     GPO        1678     A
GPO                SA4    X1+4         UI = O.UDT + 4                               GPO        1679     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                SB4    X2-2        I = N.VAR - 2                          GPO        1680    A
GPO                                                                           GPO        1681    A
GPO       CRB3     BX6    -X0*X4      RBITS[UI] = 0                          GPO        1682    A
GPO                SA6    A4                                                 GPO        1683    A
GPO                SB4    B4-1        I = I - 1                              GPO        1684    A
GPO                SA4    A4+2        UI = UI + 2                            GPO        1685    A
GPO                GT     B4,CRB3     IF I > 0                               GPO        1686    A
GPO                                                                           GPO        1687    A
GPO       CRB      ROUTINE                                                    GPO        1688    A
GPO                MX0    -UD.STCP                                           GPO        1689          I
          -FCC1504X
GPO                MX0    -UD.RFP                                            FCC1504X      9    A
GPO                SA2    UXR+1                                              GPO        1690    A
GPO                NZ     X5,CRB2     IF N.BLK > 1                           GPO        1691    A
GPO                NZ     X2,CRB2     IF UXR(2) " 0 */ EXT REF               GPO        1692    A
GPO                                                                           GPO        1693    A
GPO                SA1    O.MVL                                              GPO        1694    A
GPO                SA5    X1+1        MI = [O.MVL] + 1                       GPO        1695    A
GPO                ZR     X5,CRB      IF [MI] = 0                            GPO        1696    A
GPO                                                                           GPO        1697    A
GPO       CRB1     SA4    X5          UI = [MI]                              GPO        1698    A
GPO                SA5    A5+B1       MI = MI + 1                            GPO        1699    A
GPO                BX6    -X0*X4      RBITS[UI] = 0                          GPO        1700    A
GPO                SA6    A4                                                 GPO        1701    A
GPO                NZ     X5,CRB1     IF [MI] " 0                            GPO        1702    A
GPO                                                                           GPO        1703    A
GPO                EQ     CRB                                                GPO        1704    A
GPO       EBV      TITLE  EBV - EXPAND BIT VECTOR TO *UDI*                   GPO        1705    A
GPO      **        EBV - EXPAND BIT VECTOR TO SET BITS IN *UDI*              GPO        1706    A
GPO      *                                                                    GPO        1707    A
GPO      *         ENTRY  (X4) = BVA , BIT VECTOR ADDRESS                    GPO        1708    A
GPO      *                (X5) = BITS  , TO BE SET                           GPO        1709    A
GPO                                                                           GPO        1710    A
GPO       EBV      ROUTINE                                                    GPO        1711    A
GPO                SA4    X4-1        BI = BVA - 1                           GPO        1712    A
GPO                SA1    BVL                                                GPO        1713    A
GPO                SA2    O.UDT                                              GPO        1714    A
GPO                BX0    X5                                                 GPO        1715    A
GPO                SB6    X1+B1       L = BVL + 1                            GPO        1716    A
GPO                SB7    X2+4-120    U0 = O.UDT + 4 - 120                   GPO        1717    A
GPO                                                                           GPO        1718    A
GPO       EBV1     SB6    B6-B1       L = L - 1                              GPO        1719    A
GPO                SA4    A4+B1       BI = BI + 1;  BV = [BI]                GPO        1720    A
GPO                SB7    B7+120      U0 = U0 + 120                          GPO        1721    A
GPO                MI     B6,EBV      IF L < 0      */ END OF VECTOR         GPO        1722    A
GPO                NZ     X4,EBV2     IF BV " 0                              GPO        1723    A
GPO                PL     X4,EBV1     IF BV = +0                             GPO        1724    A
GPO                                                                           GPO        1725    A
GPO       EBV2     MX1    12                                                 GPO        1726    A
GPO                BX5    X1*X4       TB = MASK(12) & BV                     GPO        1727    A
GPO                ZR     X5,EBV4     IF TB = 0                              GPO        1728    A
GPO                                                                           GPO        1729    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                LX5    -12                                                    GPO      1730    A
GPO                MX7    13                        */ EXPAND UPPER 12 BITS      GPO      1731    A
GPO                SB3    59                                                     GPO      1732    A
GPO        EBV3    PX5    X5                                                     GPO      1733    A
GPO                NX6    B2,X5       J = NORMC( PACK(0,TB) )                    GPO      1734    A
GPO                SB4    B3-B2                                                  GPO      1735    A
GPO                AX2    B2,X7                                                  GPO      1736    A
GPO                SB5    B4+B4       K = 2*(59-J)                               GPO      1737    A
GPO                BX5    -X2*X5      TB = ^MASK(J+13) & TB                      GPO      1738    A
GPO                SA3    B7+B5       UI = U0 + K                                GPO      1739    A
GPO                BX6    X0+X3       [UI] = BITS ! [UI]                         GPO      1740    A
GPO                SA6    A3                                                     GPO      1741    A
GPO                NZ     X5,EBV3     IF TB " 0                                  GPO      1742    A
GPO                                                                             GPO      1743    A
GPO                BX4    -X1*X4      BV = ^MASK(12) & BV                        GPO      1744    A
GPO                ZR     X4,EBV1     IF BV = 0                                  GPO      1745    A
GPO                                                                             GPO      1746    A
GPO        EBV4    SX1    B1                                                     GPO      1747    A
GPO                PX5    X4                        */ EXPAND LOWER 48 BITS      GPO      1748    A
GPO                SB3    47                                                     GPO      1749    A
GPO                                                                             GPO      1750    A
GPO        EBV5    NX6    B2,X5                                                  GPO      1751    A
GPO                SB4    B3-B2       J = 47 - NORMC( PACK(0,BV) )               GPO      1752    A
GPO                SB5    B4+B4                                                  GPO      1753    A
GPO                LX2    B4,X1                                                  GPO      1754    A
GPO                SA3    B7+B5       UI = U0 + 2*J                              GPO      1755    A
GPO                IX4    X4-X2       BV = BV - SHIFT(1,J)                       GPO      1756    A
GPO                BX6    X0+X3       [UI] = BITS ! [UI]                         GPO      1757    A
GPO                PX5    X4                                                     GPO      1758    A
GPO                SA6    A3                                                     GPO      1759    A
GPO                NZ     X4,EBV5     IF BV " 0                                  GPO      1760    A
GPO                                                                             GPO      1761    A
GPO                EQ     EBV1                                                   GPO      1762    A
GPO         FUD    TITLE  FUD - FORM USE/DEF BIT VECTORS                         GPO      1763    A
GPO       **       FUD - FORM USE/DEF BIT VECTOR INFORMATION FOR A BLOCK         GPO      1764    A
GPO       *        FUD OPERATES IN TWO MODES, BUILD BIT VECTORS MODE, AND        GPO      1765    A
GPO       *        USE/DEF ONLY MODE.  COLLECTION OF USE/DEF INFORMATION FOR     GPO      1766    A
GPO       *        SCALARS WHICH ARE NOT CLASS MEMBERS IS STRAIGHTFOWARD, AND    GPO      1767    A
GPO       *        POSES NO PROBLEMS.  THE *CM* ( CLASS MEMBER ) CASE POSES MORE GPO      1768    A
GPO       *        PROBLEMS SINCE WE DONT KNOW THE SET OF VALUES THAT A USE/DEF  GPO      1769    A
GPO       *        OF A *CR* ACCESSES OR CHANGES.  LACKING THIS INFORMATION, WE  GPO      1770    A
GPO       *        MAKE THE FOLLOWING ASSUMPTIONS, WHICH ENLARGE THE LIVE EXIT   GPO      1771    A
GPO       *        AND *UBD* BIT VECTORS.                                        GPO      1772    A
GPO       *        A) A *CR* USE, USES EVERY MEMBER ( *CM* ) OF THE CLASS.       GPO      1773    A
GPO       *        B) A *CR* DEF DOESNT DEFINE ANY *CM* THAT WAS NOT EXPLICITELY GPO      1774    A
GPO       *        DEFINED, HENCE WE SHOW A USE AT THE POINTS WERE A *CR* IS     GPO      1775    A
GPO       *        DEFINED BEFORE USEAGE.                                        GPO      1776    A
GPO       *                                                                      GPO      1777    A
GPO       *        ENTRY  (X5) = [O.SEQ]                                         GPO      1778    A
GPO       *        (BBV) = 0 IF BUILDING BIT VECTORS                             GPO      1779    A
GPO       *                                                                      GPO      1780    A
GPO       *        EXIT   DEFR, ^RD, IST, STC FIELDS SET IN *UDT*                GPO      1781    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO      *         DEF, UBD AND USE BIT VECTORS SETUP IN *BVT* ( BBV = 0 )                GPO        1782     A
GPO                                                                                       GPO        1783     A
GPO       .BCM     EQU    60-UD.STCP                                                      GPO        1784     A
GPO       BCM      VFD    .BCM/-0,*P/0  BLOCK CLEAR MASK                                  GPO        1785     A
GPO                                                                                       GPO        1786     A
GPO       RDB      BFMW   UD,(DEFR,^RD,SMS) REGION DEF BITS FOR NO CODE MOTION            GPO        1787     A
GPO       FUDA     BFMW   UD,(DEFR,DEF,^RD,IST)                                           GPO        1788     A
GPO       FUDB     BFMW   UD,(DEFR,DEF,USE,^RD,IST)                                       GPO        1789     A
GPO       FUDC     BFMW   UD,(DEF,CR)                                                     GPO        1790     A
GPO       FUDD     BFMW   UD,(DEF,CM)                                                     GPO        1791     A
GPO       FUDE     BFMW   UD,(DEF,^RD,IST)                                     000140     FCC1040A      1     A
GPO                                                                                       GPO        1792     A
GPO      *         MCM - MARK CLASS MEMBERS TO SHOW A USE OF THEM                         GPO        1793     A
GPO                                                                                       GPO        1794     A
GPO       MCM      MACRO  NEXT                                                            GPO        1795     A
GPO                LOCAL  MCM1,MCM2                                                        GPO        1796     A
GPO                SB3    A0                                                              GPO        1797     A
GPO                LX6    59-UD.CRP                                                       GPO        1798     A
GPO                NZ     B3,NEXT     IF BBV " 0                                          GPO        1799     A
GPO                PL     X6,NEXT     IF ^CR[UI]                                          GPO        1800     A
GPO                LX6    1+UD.CRP-UD.ECLP                                                GPO        1801     A
GPO                SB3    X6          K = ECL[UI]                                         GPO        1802     A
GPO                ZR     B3,NEXT     IF K = 0      */ NO CLASS MEMBERS                   GPO        1803     A
GPO      *                                                                                GPO        1804     A
GPO                SX4    B1                                                              GPO        1805     A
GPO                LX4    UD.USEP                                                         GPO        1806     A
GPO                MX2    -UD.MVL                                                         GPO        1807     A
GPO                LX2    UD.MVP                                                          GPO        1808     A
GPO      *                                                                                GPO        1809     A
GPO       MCM1     SA1    B5+B3       UI = UL = UDT(K)                                    GPO        1810     A
GPO                BX6    X1+X4                                                           GPO        1811     A
GPO                SA6    A1          USE[UI] = 1                                         GPO        1812     A
GPO                BX7    -X2*X1                                                          GPO        1813     A
GPO                NZ     X7,MCM2     IF MV[UL]                                           GPO        1814     A
GPO                SX7    A1                                                              GPO        1815     A
GPO                SA7    A7+B1       MI = MI + 1;  [MI] = UI                             GPO        1816     A
GPO       MCM2     LX1    -UD.ECLP                                                        GPO        1817     A
GPO                SB3    X1          K = ECL[UI]                                         GPO        1818     A
GPO                NZ     B3,MCM1     IF K " 0                                            GPO        1819     A
GPO                ENDM                                                                   GPO        1820     A
GPO       FUD      SPACE  3,14                                                            GPO        1821     A
GPO       FUD      ROUTINE                                                                GPO        1822     A
GPO                SA1    O.UDT                                                           GPO        1823     A
GPO                SA2    X5+B1       R2 = O.SEQ + 1                                      GPO        1824     A
GPO                SB2    B1+B1                                                           GPO        1825     A
GPO                SA3    BBV                                                             GPO        1826     A
GPO                LX2    -R2.TXTLP                                                       GPO        1827     A
GPO                SB5    X1                                                              GPO        1828     A
GPO                SB6    B0          CDEF = 0                                            GPO        1829     A
GPO                MX0    -D.LSJL                                                         GPO        1830     A
GPO                IX6    X5+X2                                                           GPO        1831     A
GPO                SA0    X3+                       (A0) = BBV                            GPO        1832     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO              SA3    O.MVL                                                GPO        1833    A
GPO              SB7    X6          O.IOL = O.SEQ + TXTL[R2]  */ I/O LIST INFO  GPO        1834    A
GPO              SA4    X5+B2       DI = TBA + 2                                GPO        1835    A
GPO              MX7    0                                                       GPO        1836    A
GPO              LX0    D.LSJP                                                  GPO        1837    A
GPO              SA7    UXR         UXR = 0                                     GPO        1838    A
GPO              SA7    X3          MI = [O.MVL];  [MI] = 0 */ MARKED VAR LIST  GPO        1839    A
GPO              EQ     FUD2                                                    GPO        1840    A
GPO                                                                             GPO        1841    A
GPO      *       SCAN 1 - SCAN BLOCK FOR LD/ST/RJX"S THAT ARE CHAINED TO *UDI*  GPO        1842    A
GPO      *       AND SET BITS IN *UDT*. FORM LIST OF MARKED VARIABLES.          GPO        1843    A
GPO                                                                             GPO        1844    A
GPO      *       CHECK *UDT* ENTRY AND ADD TO MARKED LIST IF FIRST OCCURANCE    GPO        1845    A
GPO                                                                             GPO        1846    A
GPO       FUD1   LX3    D.LSJP-UD.MVP                                           GPO        1847    A
GPO              BX4    -X0*X3                                                  GPO        1848    A
GPO              NZ     X4,FUD2     IF MV[UL]     */ IF ON MARKED LIST          GPO        1849    A
GPO              SX7    A3                                                      GPO        1850    A
GPO              SA7    A7+1        MI = MI + 1;  [MI] = UL  */ SAVE *UDI* ADDR GPO        1851    A
GPO                                                                             GPO        1852    A
GPO       FUD2   SA5    A4+B2       R1 = DI + 2                                 GPO        1853    A
GPO              SA4    A5+B2       DI = R1 + 2                                 GPO        1854    A
GPO              UX6    B3,X5                                                   GPO        1855    A
GPO              BX7    -X0*X4                                                  GPO        1856    A
GPO              ZR     B3,FUD16    IF OC[R1] = 0 */ END OF BLOCK               GPO        1857    A
GPO              ZR     X7,FUD2     IF ^LSJ[DI]                                 GPO        1858    A
GPO                                                                             GPO        1859    A
GPO      *       PROCESS *LD*                                                   GPO        1860    A
GPO                                                                             GPO        1861    A
GPO              LX7    59-D.LDP                                                GPO        1862    A
GPO              PL     X7,FUD3     IF ^LD[DI]                                  GPO        1863    A
GPO              LX5    -R1.INP                                                 GPO        1864    A
GPO              SB3    X5          I = IN[R1]                                  GPO        1865    A
GPO              ZR     B3,FUD2     IF I = 0      */ INST ^CHAINED TO *UDI*     GPO        1866         I
         -FCC1088
GPO              LE     B3,B2,FUD2  IF I @ 2      */ ^ IN UDT ! LD ONLY         FCC1088      23    A
GPO              SA3    B5+B3       UI = UL = O.UDT + I                         GPO        1867    A
GPO              LX7    1+UD.USEP                                               GPO        1868    A
GPO              BX6    X7+X3       USE[UI] = 1                                 GPO        1869    A
GPO              SA6    A3                                                      GPO        1870    A
GPO              BX7    X7*X3                                                   GPO        1871    A
GPO              NZ     X7,FUD2     IF USE[UL]                                  GPO        1872    A
GPO                                                                             GPO        1873    A
GPO              MCM    FUD1        MARK CLASS REFS                             GPO        1874    A
GPO              EQ     FUD1                                                    GPO        1875    A
GPO       FUD    SPACE  2,10                                                    GPO        1876    A
GPO      *       PROCESS *ST*                                                   GPO        1877    A
GPO                                                                             GPO        1878    A
GPO       FUD3   LX7    D.LDP-D.STP                                             GPO        1879    A
GPO              PL     X7,FUD5     IF ^ST[DI]                                  GPO        1880    A
GPO              LX5    -R1.INP                                                 GPO        1881    A
GPO              SB3    X5          I = IN[R1]                                  GPO        1882    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                ZR     B3,FUD2      IF I = 0                                    GPO        1883     A
GPO                SA3    B5+B3        UI = UL = O.UDT + I                         GPO        1884     A
GPO                MX1    UD.DEFBL                                                 GPO        1885     A
GPO                LX1    UD.DEFBL+UD.DEFBP                                        GPO        1886     A
GPO                BX6    X1+X3        DEFR[UI] = DEF[UI] = 1                      GPO        1887     A
GPO                SX5    B6                                                       GPO        1888     A
GPO                LX7    1+UD.STCP                                                GPO        1889     A
GPO                IX6    X7+X6        STC[UI] = STC[UI] + 1                       GPO        1890     A
GPO                LX3    -UD.CBP                                                  GPO        1891     A
GPO                BX5    X5+X3        CDEF = CDEF ! CB[UI]                        GPO        1892     A
GPO                SA6    A3                                                       GPO        1893     A
GPO                SB6    X5                                                       GPO        1894     A
GPO                LX3    UD.CBP+59-UD.USEP                                        GPO        1895     A
GPO                MI     X3,FUD2      IF USE[UI]    */ USE BEFORE DEF             GPO        1896     A
GPO                LX7    UD.DBUP-UD.STCP                                          GPO        1897     A
GPO                BX6    X7+X6        DBU[UI] = 1                                 GPO        1898     A
GPO                LX3    1+UD.USEP                                                GPO        1899     A
GPO                SA6    A6                                                       GPO        1900     A
GPO                MCM    FUD1         SHOW USES OF CLASS MEMBERS                  GPO        1901     A
GPO                EQ     FUD1                                                     GPO        1902     A
GPO                                                                               GPO        1903     A
GPO       *        PROCESS *RJX* ( FUNCTION CALL )                                GPO        1904     A
GPO                                                                               GPO        1905     A
GPO        FUD5    LX4    59-D.TYP                                                 GPO        1906     A
GPO                PL     X4,FUD2      IF TYPE[DI] " IV                            GPO        1907     A
GPO                SA3    A4-B1        R2 = DI - 1                                 GPO        1908     A
GPO                LX3    -FI.INDXP                                                GPO        1909     A
GPO                ZR     X3,FUD2      IF [R2] = 0   */ *UJP* OR SUCH              GPO        1910     A
GPO                SA1    MRA                                                      GPO        1911     A
GPO                SA2    B7+X3        LI = O.IOL + INDX[R2] */ PARAM LIST ADDR    GPO        1912     A
GPO                UX3    B4,X3        J = LEN[R2]   */ LIST LENGTH                CCG1         79     A
GPO                LX3    FI.INDXP-FI.REGPP                                        GPO        1913     A
GPO                SX7    X3                                                       GPO        1914          I
          -CCG1
GPO                MX6    -FI.REGPL                                                CCG1         80     A
GPO                BX7    -X6*X3                                                   CCG1         81     A
GPO                BX6    X7*X1        MRA = MRA & REGP[R2]                        GPO        1915     A
GPO                LX3    FI.REGPP-FI.LENP                                         GPO        1916          I
          -CCG1
GPO                SA6    A1                                                       GPO        1917          I
          -CCG1
GPO                SB4    X3           J = LEN[R2]   */ LIST LENGTH                GPO        1918          I
          -CCG1
GPO                MX7    -FI.FTL                                                  GPO        1919          I
          -CCG1
GPO                LX3    FI.LENP-FI.FTP                                           GPO        1920          I
          -CCG1
GPO                SA6    A1                                                       CCG1         82     A
GPO                MX7    -FI.FTL                                                  CCG1         83     A
GPO                LX3    FI.REGPP-FI.FTP                                          CCG1         84     A
GPO                BX7    -X7*X3                                                   GPO        1921     A
GPO                SB3    X7                                                       GPO        1922     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                SX6    B1          XRF = 1                                  GPO         1923          I
          -FCC1254
GPO                SA6    XRF                                                  GPO         1924          I
          -FCC1254
GPO                SA3    XRF                                                  FCC1254       11     A
GPO                SX6    B1                                                   FCC1254       12     A
GPO                IX6    X3+X6       XRF = XRF + 1 */ COUNT N.EXT REFS        FCC1254       13     A
GPO                SA6    A3                                                   FCC1254       14     A
GPO                EQ     B3,B2,FUD2  IF FT[R2] = 2 */ *BEF*                   GPO         1925     A
GPO                EQ     B3,B1,FUD12 IF FT[R2] = 1 */ USER FUNCTION           GPO         1926     A
GPO                                                                            GPO         1927     A
GPO      *         PROCESS I/O APLIST                                          GPO         1928     A
GPO                                                                            GPO         1929     A
GPO                ZR     B4,FUD2     IF J = 0                                 GPO         1930          I
          -CCG1
GPO                LE     B4,FUD2     IF J @ 0                                 CCG1          85     A
GPO                LX0    UD.MVP-D.LSJP                                        GPO         1931     A
GPO                                                                            GPO         1932     A
GPO       FUD7     SA3    B5+X2       UI = O.UDT + [LI]                        GPO         1933     A
GPO                LX2    59-AP.IOP                                            GPO         1934     A
GPO                BX6    -X0*X3                                               GPO         1935     A
GPO                SB4    B4-1        J = J - 1                                GPO         1936     A
GPO                NZ     X6,FUD8     IF MV[UI]                                GPO         1937     A
GPO                SX7    A3                                                   GPO         1938     A
GPO                SA7    A7+B1       MI = MI + 1;  [MI] = UI                  GPO         1939     A
GPO                                                                            GPO         1940     A
GPO       FUD8     SX5    B1                                                   GPO         1941     A
GPO                MI     X2,FUD9     IF IO[LI] " 0 */ AN INPUT OP             GPO         1942     A
GPO                LX5    UD.USEP                                              GPO         1943     A
GPO                BX6    X5+X3       USE[UI] = 1                              GPO         1944     A
GPO                SA6    A3                                                   GPO         1945     A
GPO                EQ     FUD11                                                GPO         1946     A
GPO                                                                            GPO         1947     A
GPO       FUD9     SA1    FUDA                                                 GPO         1948     A
GPO                BX6    X1+X3       (DEFR,DEF,^RD,IST)[UI] = 1               GPO         1949     A
GPO                LX3    59-UD.USEP                                           GPO         1950     A
GPO                MI     X3,FUD10    IF USE[UI]                               GPO         1951     A
GPO                LX5    UD.DBUP     DBU[UI] = 1                              GPO         1952          I
          -FCC0789
GPO                LX2    59-AP.USEP                                           FCC0789       21     A
GPO                PL     X2,FUD9A    IF ^USE[LI]   */ ^ USE FIRST IN LIST     FCC0789       22     A
GPO                LX5    UD.USEP                                              FCC0789       23     A
GPO                BX6    X5+X6       USE[UI] = 1                              FCC0789       24     A
GPO                EQ     FUD10                                                FCC0789       25     A
GPO                                                                            FCC0789       26     A
GPO       FUD9A    LX5    UD.DBUP     DBU[UI] = 1                              FCC0789       27     A
GPO                BX6    X5+X6                                                GPO         1953     A
GPO       FUD10    SX7    B6                                                   GPO         1954     A
GPO                LX3    1+UD.USEP-UD.CBP                                     GPO         1955     A
GPO                BX4    X7+X3       CDEF = CDEF ! CB[UI]                     GPO         1956     A
GPO                SB6    X4                                                   GPO         1957     A
GPO                SA6    A3+                                                  GPO         1958     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                LX3    UD.CBP                                                GPO          1959     A
GPO                                                                             GPO          1960     A
GPO        FUD11   LX3    59-UD.USEP                                            GPO          1961     A
GPO                MI     X3,FUD11A   IF USE[UI]    */ PREVIOUS USE             GPO          1962     A
GPO                MCM    FUD11A      SHOW USES OF CLASS MEMBERS                GPO          1963     A
GPO                                                                             GPO          1964     A
GPO        FUD11A  SA2    A2+B1       LI = LI + 1                               GPO          1965     A
GPO                NZ     B4,FUD7     IF J " 0                                  GPO          1966     A
GPO                                                                             GPO          1967     A
GPO                LX0    D.LSJP-UD.MVP                                         GPO          1968     A
GPO                EQ     FUD2                                                  GPO          1969     A
GPO                                                                             GPO          1970     A
GPO      *         PROCESS USER FUNCTION                                        GPO          1971     A
GPO                                                                             GPO          1972     A
GPO        FUD12   SA6    UXR         UXR = 1       */ INDICATE COMMON VAR SPOIL GPO          1973          I
         -FCC1254
GPO                SA6    A6+1        UXR(2) = 1                                GPO          1974          I
         -FCC1254
GPO        FUD12   SX6    B1                                                    FCC1254        15     A
GPO                SA6    UXR         UXR = 1       */ INDICATE COMMON VAR SPOIL FCC1254        16     A
GPO                SA6    A6+B1       UXR(2) = 1                                FCC1254        17     A
GPO                ZR     B4,FUD2     IF J = 0      */ NO ARG LIST              GPO          1975          I
         -CCG1
GPO                LE     B4,FUD2     IF J @ 0      */ NO ARG LIST              CCG1           86     A
GPO                LX0    UD.MVP-D.LSJP                                         GPO          1976     A
GPO                SX5    B6                                                    GPO          1977     A
GPO                                                                             GPO          1978     A
GPO        FUD13   SA3    B5+X2       UI = O.UDT + [LI]                         GPO          1979          I
         -FCC1088
GPO        FUD13   SB3    X2+         I = [LI]                                  FCC1088        24     A
GPO                LE     B3,B2,FUD15 IF I @ 2      */ ^ IN UDT ! LD ONLY       FCC1088        25     A
GPO                SA3    B5+X2       UI = O.UDT + I                            FCC1088        26     A
GPO                SA1    FUDB                                                  GPO          1980     A
GPO                BX4    -X0*X3                                                GPO          1981     A
GPO                NZ     X4,FUD14    IF MV[UI]                                 GPO          1982     A
GPO                SX7    A3                                                    GPO          1983     A
GPO                SA7    A7+B1       MI = MI + 1;  [MI] = UI                   GPO          1984     A
GPO                                                                             GPO          1985     A
GPO        FUD14   BX6    X1+X3       (DEFR,DEF,USE,^RD,IST)[UI] = 1            GPO          1986     A
GPO                LX3    -UD.CBP                                               GPO          1987     A
GPO                SA2    A2+B1       LI = LI + 1                               GPO          1988          I
         -FCC0392
GPO                BX5    X3+X5       CDEF = CDEF ! CB[UI]                      GPO          1989     A
GPO                SA6    A3                                                    GPO          1990     A
GPO                LX3    UD.CBP+59-UD.USEP                                     GPO          1991     A
GPO                MI     X3,FUD15    IF USE[UL]    */ PREVIOUS USES            GPO          1992     A
GPO                MCM    FUD15       MARK CLASS MEMBERS AS USES                GPO          1993     A
GPO                                                                             GPO          1994     A
GPO        FUD15   SB4    B4-B1       J = J - 1                                 GPO          1995     A
GPO                SA2    A2+B1       LI = LI + 1                               FCC0392        88     A
GPO                NZ     B4,FUD13    IF J " 0                                  GPO          1996     A
GPO                                                                             GPO          1997     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                SB6    X5                                                    GPO        1998     A
GPO                LX0    D.LSJP-UD.MVP                                         GPO        1999     A
GPO                EQ     FUD2                                                  GPO        2000     A
GPO       FUD      SPACE  3,14                                                  GPO        2001     A
GPO      *         SCAN 2 - IF *CDEF* " 0 , CLASS REFS ( INDEXED STORES ) OR    GPO        2002     A
GPO      *         CLASS MEMBERS WERE DEFINED IN THE BLOCK. PROPAGATE THE DEFS TO GPO       2003     A
GPO      *         THE CLASS.                                                   GPO        2004     A
GPO                                                                             GPO        2005     A
GPO       FUD16    MX2    -UD.CBL                                               GPO        2006     A
GPO                SX1    B6                                                    GPO        2007     A
GPO                BX6    -X2*X1                                                GPO        2008     A
GPO                SB7    A7          ML = MI       */ END OF SCAN 1 LIST       GPO        2009     A
GPO                ZR     X6,FUD24    IF CDEF = 0                               GPO        2010     A
GPO                LX6    UD.CBP+59-UD.CRP                                      GPO        2011     A
GPO                SB6    X6                                                    GPO        2012     A
GPO                LX0    UD.MVP-D.LSJP                                         GPO        2013     A
GPO                SA2    FUDA                                                  GPO        2014     A
GPO                PL     X6,FUD20    IF ^CR[CDEF]  */ IF NO CLASS REFS         GPO        2015     A
GPO                                                                             GPO        2016     A
GPO                SA5    B7+B1       I = ML + 1                                GPO        2017     A
GPO                SA1    FUDC                                                  GPO        2018     A
GPO       FUD17    SA5    A5-B1       I = I - 1                                 GPO        2019     A
GPO                SA4    X5          UI = [I]                                  GPO        2020     A
GPO                ZR     X5,FUD20    IF [I] = 0                                GPO        2021     A
GPO                BX6    -X1+X4                                                GPO        2022     A
GPO                NZ     X6,FUD17    IF ^( DEF[UI] & CR[UI] )                  GPO        2023     A
GPO                                                                             GPO        2024     A
GPO                BX6    X2+X4       (DEFR,DEF,^RD,IST)[UI] = 1                GPO        2025     A
GPO                LX4    -UD.ECLP                                              GPO        2026     A
GPO                SB4    X4          J = ECL[UI]   */ FOLLOW CHAIN TO MEMBERS  GPO        2027     A
GPO                SA6    A4                                                    GPO        2028     A
GPO                ZR     B4,FUD17    IF J = 0      */ NO CLASS MEMBERS         GPO        2029     A
GPO                                                                             GPO        2030     A
GPO       FUD18    SA4    B5+B4       UI = O.UDT + J                            GPO        2031     A
GPO                BX3    -X0*X4                                                GPO        2032     A
GPO                NZ     X3,FUD19    IF MV[UI]                                 GPO        2033     A
GPO                SX7    A4                                                    GPO        2034     A
GPO                SA7    A7+B1       MI = MI + 1;  [MI] = UI                   GPO        2035     A
GPO                                                                             GPO        2036     A
GPO       FUD19    BX6    X2+X4       (DEFR,DEF,^RD,IST)[UI] = 1                GPO        2037     A
GPO                LX4    -UD.ECLP                                              GPO        2038     A
GPO                SB4    X4          J = ECL[UI]                               GPO        2039     A
GPO                SA6    A4                                                    GPO        2040     A
GPO                NZ     B4,FUD18    IF J " 0                                  GPO        2041     A
GPO                EQ     FUD17                                                 GPO        2042     A
GPO       FUD      SPACE  2,10                                                  GPO        2043     A
GPO      *         SEARCH  MARKED VARS LIST FOR CLASS MEMBERS THAT WERE DEFINED GPO        2044     A
GPO      *         AND PROPAGATE THE DEF TO THE *CLASS REF* MEMBER.             GPO        2045     A
GPO                                                                             GPO        2046     A
GPO       FUD20    ZR     B6,FUD24    IF CM[CDEF] = 0  */ NO CLASS MEMBERS DEFINE GPO        2047     A
GPO                SA1    FUDD                                                  GPO        2048     A
GPO                SA2    FUDE                                        000160    FCC1040A      2     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                SA5    B7+B1       I = ML + 1                                          GPO        2049     A
GPO                                                                                       GPO        2050     A
GPO       FUD21    SA5    A5-B1       I = I - 1                                           GPO        2051     A
GPO                SA4    X5          UI = [I]                                            GPO        2052     A
GPO                ZR     X5,FUD24    IF [I] = 0    */ END OF LIST                        GPO        2053     A
GPO                BX6    -X1+X4                                                          GPO        2054     A
GPO                NZ     X6,FUD21    IF ^( DEF[UI] & CM[UI] )                            GPO        2055     A
GPO                LX4    -UD.BMIP                                                        GPO        2056     A
GPO                SA3    B5+X4       UI = O.UDT + BMI[UI]  */ BASE MEMBER                GPO        2057     A
GPO                BX4    -X0*X3                                                          GPO        2058     A
GPO                NZ     X4,FUD22    IF MV[UI]                                           GPO        2059     A
GPO                SX7    A3                                                              GPO        2060     A
GPO                SA7    A7+B1       MI = MI + 1;  [MI] = UI                             GPO        2061     A
GPO                                                                                       GPO        2062     A
GPO       FUD22    BX6    X2+X3       (DEFR,DEF,^RD,IST)[UI] = 1                          GPO        2063          I
         -FCC1040A
GPO       FUD22    BX6    X2+X3       (DEF,^RD,IST)[UI] = 1                    000180     FCC1040A      3     A
GPO                SA6    A3                                                              GPO        2064     A
GPO                EQ     FUD21                                                           GPO        2065     A
GPO                                                                                       GPO        2066     A
GPO      *         TERMINATE MARKED VAR LIST AND FORM BIT VECTORS ( BBV = 0 )              GPO        2067     A
GPO                                                                                       GPO        2068     A
GPO       FUD24    SA1    BN                                                              GPO        2069     A
GPO                SA2    O.BIT                                                           GPO        2070     A
GPO                SA3    BBV                                                             GPO        2071     A
GPO                SB2    X1                                                              GPO        2072     A
GPO                SA4    X2+B2       BIW = BIT(BN)                                       GPO        2073     A
GPO                MX7    0                                                               GPO        2074     A
GPO                SA7    A7+B1       MI = MI + 1;  [MI] = 0                              GPO        2075     A
GPO                NZ     X3,FUD35    IF BBV " 0    */ NOT BUILDING BIT VECTORS           GPO        2076     A
GPO                                                                                       GPO        2077     A
GPO                SA5    BVL                                                             GPO        2078     A
GPO                SA3    L.BVT                                                           GPO        2079     A
GPO                SA2    O.BVT                                                           GPO        2080     A
GPO                SB6    X3          OL = L.BVT                                          GPO        2081     A
GPO                LX3    BI.BVIP                                                         GPO        2082     A
GPO                SX1    X5+B1                                                           GPO        2083     A
GPO                BX6    X3+X4       BVI[BIT(BN)] = L.BVT                                GPO        2084     A
GPO                LX1    2                                                               GPO        2085     A
GPO                SA6    A4                                                              GPO        2086     A
GPO                LX3    -BI.BVIP                                                        GPO        2087     A
GPO                IX7    X1+X3       L.BVT = L.BVT + 4*(BVL+1)                           GPO        2088     A
GPO                SA7    A3                                                              GPO        2089     A
GPO                                                                                       GPO        2090     A
GPO                IF     DEF,/DEBUG/FUD,2                                                GPO        2091     A
GPO                SX6    X3                                                              GPO        2092     A
GPO                SA6    BVI                                                             GPO        2093     A
GPO                SB7    X5+B1       VL = BVL + 1  */ VECTOR LENGTH                      GPO        2094     A
GPO                SA1    O.MVL                                                           GPO        2095     A
GPO                SB6    X2+B6       BI = O.BVT + OL  */ *BVT* ADDRESS                   GPO        2096     A
GPO                SX0    B1                                                              GPO        2097     A
GPO                NZ     X5,FUD30    IF BVL " 0                                          GPO        2098     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO       FUD      SPACE  2,10                                                    GPO        2099     A
GPO      *         SETUP BIT VECTORS FOR PROGRAM WITH @ 60 REFERENCED VARIABLES    GPO        2100     A
GPO                                                                                GPO        2101     A
GPO                MX6    0           DV = 0                                       GPO        2102     A
GPO                SA3    X1+B1       MI = O.MVL + 1                               GPO        2103     A
GPO                BX5    X5-X5       DUV = 0                                      GPO        2104     A
GPO                MX4    0           UV = 0                                       GPO        2105     A
GPO                ZR     X3,FUD26    IF [MI] = 0   */ NO REFED VARS IN BLOCK      GPO        2106     A
GPO                SA2    X3+B1       U2 = [MI] + 1                                GPO        2107     A
GPO                SA1    X3          UW = [MI]                                    GPO        2108     A
GPO                                                                                GPO        2109     A
GPO       FUD25    UX7    B2,X2       J = BITN[U2]                                 GPO        2110     A
GPO                LX1    -UD.DEFP                                                 GPO        2111     A
GPO                BX2    X0*X1                                                    GPO        2112     A
GPO                LX7    B2,X2                                                    GPO        2113     A
GPO                BX6    X7+X6       DV = DV ! SHIFT( DEF[UW] , J )               GPO        2114     A
GPO                LX1    UD.DEFP-UD.DBUP                                          GPO        2115     A
GPO                BX2    X0*X1                                                    GPO        2116     A
GPO                LX7    B2,X2                                                    GPO        2117     A
GPO                SA3    A3+B1       MI = MI + 1                                  GPO        2118     A
GPO                BX5    X7+X5       DUV = DUV ! SHIFT( DBU[UW] , J )             GPO        2119     A
GPO                LX1    UD.DBUP-UD.USEP                                          GPO        2120     A
GPO                BX2    X0*X1                                                    GPO        2121     A
GPO                LX7    B2,X2                                                    GPO        2122     A
GPO                SA2    X3+B1       U2 = [MI] + 1                                GPO        2123     A
GPO                BX4    X7+X4       UV = UV ! SHIFT( USE[UW] , J )               GPO        2124     A
GPO                SA1    X3          UW = [MI]                                    GPO        2125     A
GPO                NZ     X3,FUD25    IF [MI] " 0                                  GPO        2126     A
GPO                                                                                GPO        2127     A
GPO                SA1    UXR                                                      GPO        2128          I
          -FCC0716
GPO                ZR     X1,FUD26    IF UXR = 0                                   GPO        2129          I
          -FCC0716
GPO       FUD26    SA1    UXR                                                      FCC0716       6     A
GPO                ZR     X1,FUD26A   IF UXR = 0                                   FCC0716       7     A
GPO                SA2    O.BVT                                                    GPO        2130     A
GPO                SA3    X2          SV = [O.BVT]  */ COMMON VAR SPOIL VECTOR     GPO        2131     A
GPO                BX6    X3+X6       DV = SV ! DV                                 GPO        2132     A
GPO                BX5    -X3*X5      DUV = ^SV & DUV                              GPO        2133     A
GPO                BX4    X3+X4       UV = SV ! UV                                 GPO        2134     A
GPO                                                                                GPO        2135     A
GPO       FUD26    SA6    B6          [BI] = DV                                    GPO        2136          I
          -FCC0716
GPO       FUD26A   SA6    B6          [BI] = DV                                    FCC0716       8     A
GPO                BX7    -X5*X4                                                   GPO        2137     A
GPO                SA7    B6+B1       [BI+1] = ^DUV & UV  */ UBD                   GPO        2138     A
GPO                BX6    X4                                                       GPO        2139     A
GPO                SA6    A7+B1       [BI+2] = UV                                  GPO        2140     A
GPO                SX7    0                                                        GPO        2141     A
GPO                SA7    A6+B1       [BI+3] = 0    */ LX = 0                      GPO        2142     A
GPO                                                                                GPO        2143     A
GPO      *         CLEAR BLOCK RELATIVE BITS IN *UDT*                              GPO        2144     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                                                                    GPO      2145    A
GPO       FUD27    BSS    0                                           GPO      2146    A
GPO                IF     DEF,/DEBUG/FUD,1                            GPO      2147    A
GPO                DCALL  PRNTBV,([O.BVT],BVI,=3)                     GPO      2148    A
GPO                                                                   GPO      2149    A
GPO       FUD28    SA1    O.MVL                                       GPO      2150    A
GPO                SA4    BCM                                         GPO      2151    A
GPO                SA5    X1+1        MI = O.MVL + 1                  GPO      2152    A
GPO                ZR     X5,FUD      IF [MI] = 0                     GPO      2153    A
GPO                                                                   GPO      2154    A
GPO       FUD29    SA2    X5          UW = [MI]                       GPO      2155    A
GPO                SA5    A5+B1       MI = MI + 1                     GPO      2156    A
GPO                BX6    -X4*X2      BCM[UW] = 0   */ CLEAR BLOCK / REGION BITS  GPO  2157  A
GPO                SA6    A2                                          GPO      2158    A
GPO                NZ     X5,FUD29    IF [MI] " 0                     GPO      2159    A
GPO                                                                   GPO      2160    A
GPO                EQ     FUD                                         GPO      2161    A
GPO       FUD      SPACE  3,30                                        GPO      2162    A
GPO      *         SETUP BIT VECTORS FOR PROGRAM WITH N.VAR > 60      GPO      2163    A
GPO      *         FIRST CHAIN *MVL* ENTRIES ON THEIR WORD INDEX      GPO      2164    A
GPO                                                                   GPO      2165    A
GPO       FUD30    SB5    B7-B1                                       GPO      2166    A
GPO                SX6    B7+B7                                       GPO      2167    A
GPO                SB4    B6+B7                                       GPO      2168    A
GPO                SA0    X6+B4       BT = BI + 3*VL  */ BUCKET TABLE GPO      2169    A
GPO                MX7    0                                           GPO      2170    A
GPO                SA3    X1+B1       MI = O.MVL + 1                  GPO      2171    A
GPO                SB2    A0                                          GPO      2172    A
GPO                SA7    A0                                          GPO      2173    A
GPO                                                                   GPO      2174    A
GPO      +         SB5    B5-B1       FOR I = 0 TO VL-1; BT(I) = 0    GPO      2175    A
GPO                SA7    A7+B1                                       GPO      2176    A
GPO                GT     B5,*                                        GPO      2177    A
GPO                                                                   GPO      2178    A
GPO                SA2    UXR                                         GPO      2179    A
GPO                SA5    O.BVT                                       GPO      2180    A
GPO                SB3    X2                                          GPO      2181    A
GPO                SB4    B0          WN = 0        */ WORD NUMBER ( 0,1,... )  GPO  2182  A
GPO                SB5    X5                                          GPO      2183    A
GPO                ZR     X3,FUD32    IF [MI] = 0   */ NO REFED VARS IN BLOCK  GPO  2184  A
GPO                                                                   GPO      2185    A
GPO       FUD31    SA4    X3+B1       U2 = [MI] + 1;  W = WI[U2]      GPO      2186    A
GPO                SA5    B2+X4                                       GPO      2187    A
GPO                SX7    A3                                          GPO      2188    A
GPO                LX5    18                                          GPO      2189    A
GPO                BX6    X5+X3       LINK[MI] = BT(W)  */ LINK THIS TO LAST  GPO  2190  A
GPO                SA6    A3                                          GPO      2191    A
GPO                SA7    A5          BT(W) = MI    */ POINT BUCKET TO THIS  GPO  2192  A
GPO                SA3    A3+B1       MI = MI + 1                     GPO      2193    A
GPO                NZ     X3,FUD31    IF [MI] " 0                     GPO      2194    A
GPO                                                                   GPO      2195    A
GPO      *         NOW FORM THE BIT VECTORS                           GPO      2196    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                                                                                    GPO        2197    A
GPO       FUD32   SA2    A0          LP = [BT]     */ LIST POINTER                     GPO        2198    A
GPO               MX6    0           DV = 0                                            GPO        2199    A
GPO               BX5    X5-X5       DUV = 0                                           GPO        2200    A
GPO               SA3    X2          MI = [LP]                                         GPO        2201    A
GPO               MX4    0           UV = 0                                            GPO        2202    A
GPO               SA0    A0+B1       BT = BT + 1                                       GPO        2203    A
GPO               ZR     X2,FUD34    IF LP = 0     */ LIST EMPTY FOR THIS WORD         GPO        2204    A
GPO                                                                                    GPO        2205    A
GPO       FUD33   SA2    X3+B1       U2 = [MI] + 1                                     GPO        2206    A
GPO               SA1    X3          UW = [MI]                                         GPO        2207    A
GPO               AX3    18          L = LINK[MI]                                      GPO        2208    A
GPO               UX7    B2,X2       J = BITN[U2]                                      GPO        2209    A
GPO               LX1    -UD.DEFP                                                      GPO        2210    A
GPO               BX2    X0*X1                                                         GPO        2211    A
GPO               LX7    B2,X2                                                         GPO        2212    A
GPO               BX6    X7+X6       DV = DV ! SHIFT( DEF[UW] , J )                    GPO        2213    A
GPO               LX1    UD.DEFP-UD.DBUP                                               GPO        2214    A
GPO               BX2    X0*X1                                                         GPO        2215    A
GPO               LX7    B2,X2                                                         GPO        2216    A
GPO               BX5    X7+X5       DUV = DUV ! SHIFT( DBU[UW] , J )                  GPO        2217    A
GPO               LX1    UD.DBUP-UD.USEP                                               GPO        2218    A
GPO               BX2    X0*X1                                                         GPO        2219    A
GPO               LX7    B2,X2                                                         GPO        2220    A
GPO               SX1    X3                                                            GPO        2221    A
GPO               BX4    X7+X4       UV = UV ! SHIFT( USE[UW] , J )                    GPO        2222    A
GPO               SA3    X3          MI = [L]      */ NEXT ON CHAIN                    GPO        2223    A
GPO               NZ     X1,FUD33    IF L " 0                                          GPO        2224    A
GPO                                                                                    GPO        2225    A
GPO       FUD34   ZR     B3,FUD34A   IF UXR = 0                                        GPO        2226    A
GPO               SA3    B5+B4       SV = BVT(WN)                                      GPO        2227    A
GPO               BX6    X3+X6       DV = SV ! DV                                      GPO        2228    A
GPO               BX5    -X3*X5      DUV = ^SV & DUV                                   GPO        2229    A
GPO               BX4    X3+X4       UV = SV ! UV                                      GPO        2230    A
GPO                                                                                    GPO        2231    A
GPO       FUD34A  SA6    B6+B4       SA = BI + WN;  [SA] = DV                          GPO        2232    A
GPO               BX7    -X5*X4                                                        GPO        2233    A
GPO               SB4    B4+B1       WN = WN + 1                                       GPO        2234    A
GPO               SA7    A6+B7       SA = SA + VL;  [SA] = ^DUV & UV  */ UBD           GPO        2235    A
GPO               BX6    X4                                                            GPO        2236    A
GPO               SA6    A7+B7       SA = SA + VL;  [SA] = UV                          GPO        2237    A
GPO               MX7    0                                                             GPO        2238    A
GPO               SA7    A6+B7       SA = SA + VL;  [SA] = 0  */ LX = 0                GPO        2239    A
GPO               LT     B4,B7,FUD32 IF WN < VL                                        GPO        2240    A
GPO                                                                                    GPO        2241    A
GPO               EQ     FUD27                                                         GPO        2242    A
GPO       FUD     EJECT                                                                GPO        2243    A
GPO      *        BBV " 0 , IF PROGRAM BLOCK , THEN CLEAR BLOCK BITS                   GPO        2244    A
GPO                                                                                    GPO        2245    A
GPO       FUD35   BSS    0                                                             GPO        2246    A
GPO               IF     DEF,/DEBUG/FUD,1                                              GPO        2247    A
GPO               DCALL  PRNTUDI,([O.UDT],L.UDT)   PRINT BITS SET                      GPO        2248    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                                                                     GPO        2249     A
GPO                LX4    59-BI.RBP                                    GPO        2250     A
GPO                MI     X4,FUD28    IF RB[BIW]    */ PROGRAM BLOCK   GPO        2251     A
GPO                                                                     GPO        2252     A
GPO      *         PROCESS HOLDING BLOCK, PROPAGATE OUT INFORMATION FROM INNER   GPO  2253  A
GPO      *         LOOP.  SET *DEFR, ^RD, IST* BITS FOR ALL VARIABLES DEFINED    GPO  2254  A
GPO      *         IN THE INNER LOOP.  ACCUMULATE THE LOOP USEAGE VECTOR FOR GRA.  GPO 2255  A
GPO      *         NOTE THAT THE LOOP WE ARE PROCESSING MUST CONTAIN AT LEAST    GPO  2256  A
GPO      *         2 BLOCKS, SO *CRB* WILL CLEAR ALL OF *UDT*                    GPO  2257  A
GPO                                                                     GPO        2258     A
GPO                SA5    O.SEQ       T = [O.SEQ]                      GPO        2259     A
GPO                SA1    N.HB                                         GPO        2260     A
GPO                SA5    X5+B1       R2 = [T+1]    */ R2 OF *BOS*     GPO        2261     A
GPO                SX7    B1                                           GPO        2262     A
GPO                SX6    X1+B1       N.HB = N.HB + 1                  GPO        2263     A
GPO                LX5    59-R2.UXRP                                   GPO        2264     A
GPO                SA6    A1                                           GPO        2265     A
GPO                PL     X5,FUD36    IF ^UXR[R2]   */ NO USER EXT REFS IN LP   GPO  2266  A
GPO                SA7    UXR+1       UXR(2) = 1                       GPO        2267     A
GPO                                                                     GPO        2268     A
GPO       FUD36    SA1    MRA                                          GPO        2269     A
GPO                LX5    1+R2.UXRP-R2.MRAP                            GPO        2270     A
GPO                BX6    X1*X5       MRA = MRA & MRA[R2]              GPO        2271     A
GPO                SA6    A1                                           GPO        2272     A
GPO                SA1    MRA.X                                        FCC1254      18     A
GPO                BX6    X1*X5       MRA.X = MRA.X & MRA[R2]          FCC1254      19     A
GPO                SA6    A1                                           FCC1254      20     A
GPO                LX4    1+BI.RBP-BI.PBNP                             GPO        2273     A
GPO                SB3    X4          HN = PBN[BIW] */ BN OF HEADER    GPO        2274     A
GPO                SB5    B0          I = 0                            GPO        2275     A
GPO                FBVA   B3          BVA = O.BVT + BVI[BIT(HN)]       GPO        2276     A
GPO                SA5    LUV                                          GPO        2277     A
GPO                SB6    X7+B2       RUA = BVA + VL  */ *LU* VECTOR OF LOOP   GPO  2278  A
GPO       FUD37    SA1    X5+B5                                        GPO        2279     A
GPO                SA2    B6+B5       LUV = LUV ! [RUA]                GPO        2280     A
GPO                SB5    B5+B1                                        GPO        2281     A
GPO                BX6    X1+X2                                        GPO        2282     A
GPO                SA6    A1                                           GPO        2283     A
GPO                LT     B5,B2,FUD37                                  GPO        2284     A
GPO                                                                     GPO        2285     A
GPO                SX4    X7                                           GPO        2286     A
GPO                SA5    RDB                                          GPO        2287     A
GPO                RJ     EBV         SET REGION DEF BITS IN *UDT*     GPO        2288     A
GPO                                                                     GPO        2289     A
GPO                SA5    O.SEQ                                        GPO        2290     A
GPO                RJ     CHB         CHECK HOLDING BLOCK FOR COMBINE WITH PRED   GPO  2291  A
GPO                EQ     FUD28                                        GPO        2292     A
GPO        CHB     TITLE  CHB - COMBINE HOLDING BLOCK WITH IMMEDIATE PRED   GPO  2293  A
GPO      **        CHB - COMBINE HOLDING BLOCK WITH IMMEDIATE PRED      GPO        2294     A
GPO      *         COALESCE THE HOLDING BLOCK WITH ITS IMMEDIATE PREDECESSOR SO  GPO  2295  A
GPO      *         INITIALIZATION CODE FROM INTEGER POLYNOMIALS IS REMOVED OF    GPO  2296  A
GPO      *         ITS DEPENDENCY ON THE INNER LOOP INDUCTION VARIABLE, AND IS   GPO  2297  A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO      *         SUBJECT TO FURTHER MOTION, ETC.                                  GPO        2298     A
GPO      *                                                                          GPO        2299     A
GPO      *         ENTRY  (X5) = [O.SEQ]                                            GPO        2300     A
GPO      *                (BSI) = *BST* INDEX OF HOLDING BLOCK                      GPO        2301     A
GPO                                                                                 GPO        2302     A
GPO       CHB      ROUTINE                                                          GPO        2303     A
GPO                SA1    L.SEQ                                                     GPO        2304     A
GPO                SB5    X1-8                                                      GPO        2305     A
GPO                LE     B5,CHB      IF L.SEQ @ 8  */ EMPTY BLOCK                  GPO        2306          I
          -FCC0602
GPO                SB4    4                                                         FCC0602       5     A
GPO                LE     B5,B4,CHB   IF L.SEQ @ 12 */ EMPTY OR *LAB* ONLY BLOCK   FCC0602       6     A
GPO                                                                                 GPO        2307     A
GPO      *         CHECK CURRENT BLOCK                                              GPO        2308     A
GPO                                                                                 GPO        2309     A
GPO                SA2    O.BIT                                                     GPO        2310     A
GPO                SA3    X5          R1 = [[O.SEQ]]                                GPO        2311     A
GPO                SB2    X2                                                        GPO        2312     A
GPO                SA4    B2+X3       BIW = BIT(R1)                                 GPO        2313     A
GPO                LX4    -BI.BVIP                                                  GPO        2314     A
GPO                SX7    X4          VI2 = BVI[BIW]                                GPO        2315     A
GPO                LX4    59-BI.RBP+BI.BVIP                                         GPO        2316     A
GPO                MI     X4,CHB      IF RB[BIW]    */ NOT A *HB*                   GPO        2317     A
GPO                LX4    BI.RBP-BI.ILP                                             GPO        2318          I
          -FCC0602
GPO                MI     X4,CHB      IF IL[BIW]    */ MULT PRED IF LOOP            GPO        2319          I
          -FCC0602
GPO                                                                                 GPO        2320          I
          -FCC0602
GPO      *         USE THE INTERVAL INFORMATION IN *BST* TO FIND THE PRED OF        GPO        2321          I
          -FCC0602
GPO      *         THE *HB*.  SINCE IT IS ^IL , IT MUST HAVE 1 PRED AND THE EDGE    GPO        2322          I
          -FCC0602
GPO      *         IS ^JP .                                                         GPO        2323          I
          -FCC0602
GPO                                                                                 GPO        2324          I
          -FCC0602
GPO                                                                                 FCC0602       7     A
GPO                SA1    O.BST                                                     GPO        2325     A
GPO                SA2    BSI                                                       GPO        2326     A
GPO                SA0    A4          B = BIT + R1  */ SAVE BIT ADDR OF *HB*        FCC0841       5     A
GPO                SA3    EIT                                                       GPO        2327     A
GPO                SB3    X1+B1                     (B3) = BST+1                    GPO        2328     A
GPO                SB4    X2          I = BSI                                       GPO        2329     A
GPO                SB6    X3                                                        GPO        2330     A
GPO                SA5    B3+B4       IPW = BST(I+1)                                GPO        2331     A
GPO                LX5    -IM.BNP                                                   GPO        2332     A
GPO                SA1    B6+X5       EIW = EIT + BN[IPW]                           GPO        2333     A
GPO                LX1    -EI.PRIP                                                  GPO        2334     A
GPO                UX1    B7,X1                                                     FCC0602       8     A
GPO                NE     B7,B1,CHB   IF NP[EIW] " 1  */ MORE THAN 1 PRED           FCC0602       9     A
GPO                SA3    B6+X1       PIW = EIT + PRI[EIW]                          GPO        2335     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO              LX3    -ET.PREDP                                          GPO        2336    A
GPO              SA4    B6+X3       EIWP = EIT + PRED[PIW]                 GPO        2337    A
GPO              LX4    -EI.SUIP                                           GPO        2338    A
GPO              UX4    B7,X4                                              GPO        2339    A
GPO              GT     B7,B1,CHB   IF NS[EIWP] > 1  */ MORE THAN 1 SUCC   GPO        2340    A
GPO              SB6    X3          PBN = PRED[PIW]  */ BN OF PRED         GPO        2341    A
GPO                                                                        GPO        2342    A
GPO      *       SEARCH *BST* FOR PRED OF *HB*                             GPO        2343    A
GPO                                                                        GPO        2344    A
GPO       CHB0   SB4    B4-2        I = I - 2                              GPO        2345    A
GPO              ZR     B4,CHB      IF I = 0      */ END OF *BST*          GPO        2346    A
GPO              SA4    B3+B4       IPW = BST(I+1)                         GPO        2347    A
GPO              LX4    -IM.BNP                                            GPO        2348    A
GPO              SB7    X4                                                 GPO        2349    A
GPO              NE     B7,B6,CHB0  IF BN[IPW] " PBN                       GPO        2350    A
GPO                                                                        GPO        2351    A
GPO              LX4    IM.BNP-IM.BIP                                      GPO        2352    A
GPO              SX3    B4                                                 GPO        2353    A
GPO              SA5    B2+X4       BIWP = BIT(BI[IPW])                    GPO        2354    A
GPO              LX3    30                                                 GPO        2355    A
GPO              BX6    X3+X2       BSIS = SHIFT(I,30) ! BSI               GPO        2356    A
GPO              SA6    BSIS                                               GPO        2357    A
GPO              LX5    59-BI.RBP                                          GPO        2358    A
GPO              PL     X5,CHB      IF ^RB[BIWP]  */ PRED IS A *HB*        GPO        2359    A
GPO              BX6    X5                                                 FCC0602      10    A
GPO              SA6    A6+B1                                              FCC0602      11    A
GPO              SA2    A0          BIW = [B]     */ REFETCH BIT WORD      FCC0841       6    A
GPO              LX2    59-BI.RBP                                          FCC0841       7    A
GPO              BX6    X2-X5                                              FCC0841       8    A
GPO              LX6    BI.RBP-BI.FJP                                      FCC0841       9    A
GPO              MI     X6,CHB      IF FJ[BIW] " FJ[BIWP]                  FCC0841      10    A
GPO                                                                        GPO        2360    A
GPO      *       MERGE THE BLOCKS                                          GPO        2361         I
         -CCGA021
GPO                                                                        GPO        2362         I
         -CCGA021
GPO              SA7    VI2                                                GPO        2363    A
GPO              SX7    X5          VI1 = BVI[BIWP]                        GPO        2364    A
GPO              SA7    A7+B1                                              GPO        2365    A
GPO              ALLOC  MOD,B5      ALLOC( MOD , L.SEQ-8 )                 GPO        2366         I
         -CCGA021
GPO                                                                        CCGA021      39    A
GPO      *       SAVE ELIMINATED LABEL DEF INFO IN *ELT* FOR USE BY *WII*  CCGA021      40    A
GPO                                                                        CCGA021      41    A
GPO              LX2    BI.RBP-BI.ILP                                      CCGA021      42    A
GPO              PL     X2,CHB0A    IF IL[B]      */ NO LABEL DEF          CCGA021      43    A
GPO                                                                        CCGA021      44    A
GPO              SA1    O.SEQ                                              CCGA021      45    A
GPO              SA2    O.ELT                                              CCGA021      46    A
GPO              SA4    X1+4        R1 = [SEQ+4]  */ IH OF PROG LABEL      CCGA021      47    A
GPO              SA3    A4+B1       R2 = [SEQ+5]  */ IH OF GL              CCGA021      48    A
GPO              SX4    X4                                                 CCGA021      49    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                SX3    X3-I.GL                                                  CCGA021      50     A
GPO                LX4    18                                                       CCGA021      51     A
GPO                SA1    L.ELT                                                    CCGA021      52     A
GPO                SB2    X2+B1                                                    CCGA021      53     A
GPO                BX6    X3+X4                                                    CCGA021      54     A
GPO                SA6    B2+X1       ELT(L.ELT+1) = 24/,18/IH-PROG,18/H-GL        CCGA021      55     A
GPO                SX6    X1+B1       L.ELT = L.ELT + 1                            CCGA021      56     A
GPO                SA6    A1                                                       CCGA021      57     A
GPO                                                                               CCGA021      58     A
GPO       *        MERGE THE BLOCKS                                                CCGA021      59     A
GPO                                                                               CCGA021      60     A
GPO        CHB0A   ALLOC  MOD,B5      ALLOC( MOD, L.SEQ-8 )                        CCGA021      61     A
GPO                SX3    X2                                                       GPO        2367     A
GPO                SA2    O.SEQ                                                    GPO        2368     A
GPO                MOVE   X1+B6,X2,X3 MOVE( L.SEQ-4 , O.SEQ , O.MOD )              GPO        2369     A
GPO                                                                               GPO        2370     A
GPO                CALL   RBS         RELEASE THE BLOCK                            GPO        2371          I
          -CCGNC
GPO                CALL   RBS#        RELEASE THE BLOCK                            CCGNC       630     A
GPO                SA4    BSIS                                                     GPO        2372     A
GPO                LX4    30                                                       GPO        2373     A
GPO                SX6    X4          BSI = SETX( SHIFT(BSIS,30) )                 GPO        2374     A
GPO                SA6    BSI                                                      GPO        2375     A
GPO                CALL   RTB         GET PREDECESSOR BLOCK                        GPO        2376          I
          -CCGNC
GPO                CALL   RTB#        GET PREDECESSOR BLOCK                        CCGNC       631     A
GPO                                                                               GPO        2377     A
GPO       *        SETUP *MOD* CONTROL WORD                                        GPO        2378     A
GPO                                                                               GPO        2379     A
GPO                SA5    O.SEQ                                                    GPO        2380     A
GPO                SA3    L.MOD                                                    GPO        2381     A
GPO                SX1    4                                                        GPO        2382          I
          -FCC0602
GPO                SX1    4           K = 4                                        FCC0602      12     A
GPO                MX6    0           DEL = 0                                      FCC0602      13     A
GPO                SA4    X5+B1       R2 = [O.SEQ+1]                               GPO        2383     A
GPO                LX4    -R2.TXTLP                                                GPO        2384     A
GPO                SX2    X4-8        II = TXTL[R2] - 8                            GPO        2385     A
GPO                IX3    X3-X1                                                    GPO        2386          I
          -FCC0602
GPO                SA5    BSIS+1                                                   FCC0602      14     A
GPO                LX5    BI.RBP-BI.FJP                                            FCC0602      15     A
GPO                PL     X5,CHB1A    IF ^FJ[BIWP]                                 FCC0602      16     A
GPO                MX6    1           DEL = 1       */ SET TO DELETE UJP AND LAB   FCC0602      17     A
GPO                SX1    8           K = 8                                        FCC0602      18     A
GPO        CHB1A   IX3    X3-X1                                                    FCC0602      19     A
GPO                LX1    ML.MTIP                                                  GPO        2387     A
GPO                LX2    ML.IIP                                                   GPO        2388     A
GPO                LX3    ML.NIP-2                                                 GPO        2389     A
GPO                BX4    X2+X3                                                    GPO        2390     A
GPO                IX1    X4+X1                                                    GPO        2391     A
GPO                ADDWRD MLT,X1      ADDWRD( MLT , MCW(0,II,(L.MOD-4)/4,4))       GPO        2392          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
-FCC0602
GPO              BX1    X6+X1                                          FCC0602      20     A
GPO              ADDWRD MLT,X1      ADDWRD( MLT , MCW(DEL,II,(L.MOD-K)/4,K) )   FCC0602      21     A
GPO              MX6    59                                             GPO        2393          I
-FCC0509
GPO              CALL   MPB         MERGE, SQUEEZE AND REWRITE THE BLOCK        GPO        2394          I
-FCC0509
GPO                                                                    GPO        2395     A
GPO      *       NOW ADJUST THE BIT VECTORS OF BOTH BLOCKS             GPO        2396     A
GPO      *       USE1 = USE1 ! USE2 , DEF1 = DEF1 ! DEF2               GPO        2397     A
GPO      *       UBD1 = UBD1 ! (^DEF1&UBD2) , LX1 = LX2                GPO        2398     A
GPO                                                                    GPO        2399     A
GPO              SA2    VI2                                            GPO        2400     A
GPO              SA3    VL                                             GPO        2401     A
GPO              SA4    O.BVT                                          GPO        2402     A
GPO              SA1    A2+B1                                          GPO        2403     A
GPO              SB3    X3          L = VL                             GPO        2404     A
GPO              SB7    X4                                             GPO        2405     A
GPO              SB6    B7+X1       V1A = O.BVT + VI1                  GPO        2406     A
GPO              SB7    B7+X2       V2A = O.BVT + VI2                  GPO        2407     A
GPO              SB2    B0          I = 0                              GPO        2408     A
GPO              SB4    B3+B3                                          GPO        2409     A
GPO                                                                    GPO        2410     A
GPO       CHB1   SA1    B6+B2                                          GPO        2411     A
GPO              SA2    B7+B2                                          GPO        2412     A
GPO              SA3    A1+B4                                          GPO        2413     A
GPO              SA4    A2+B4                                          GPO        2414     A
GPO              SB2    B2+B1       I = I + 1                          GPO        2415     A
GPO              BX6    X1+X2       DEF1 = DEF1 ! DEF2                 GPO        2416     A
GPO              SA6    A1                                             GPO        2417     A
GPO              MX7    0                                              GPO        2418     A
GPO              SA7    A2          DEF2 = 0                           GPO        2419     A
GPO              BX6    X3+X4       USE1 = USE1 ! USE2                 GPO        2420     A
GPO              SA6    A3                                             GPO        2421     A
GPO              SA7    A4          USE2 = 0                           GPO        2422     A
GPO              LT     B2,B3,CHB1  IF I < L                           GPO        2423     A
GPO                                                                    GPO        2424     A
GPO              SB2    B0          I = 0                              GPO        2425     A
GPO              SB7    B7+B3                                          GPO        2426     A
GPO                                                                    GPO        2427     A
GPO       CHB2   SA2    B6+B2       DEF1 = V1A(I)                      GPO        2428     A
GPO              SA1    A2+B3       UBD1 = V1A(I+VL)                   GPO        2429     A
GPO              SA3    B7+B2                                          GPO        2430     A
GPO              SA4    A3+B4                                          GPO        2431     A
GPO              BX5    -X2*X3                                         GPO        2432     A
GPO              BX6    X1+X5       UBD1 = UBD1 ! (^DEF1 & UDB2)       GPO        2433     A
GPO              SA6    A1                                             GPO        2434     A
GPO              SB2    B2+B1       I = I + 1                          GPO        2435     A
GPO              BX7    X4          LX1 = LX2                          GPO        2436     A
GPO              SA7    A1+B4                                          GPO        2437     A
GPO              LT     B2,B3,CHB2  IF I < L                           GPO        2438     A
GPO                                                                    GPO        2439     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

GPO                SX5    B6                                                GPO          2440     A
GPO                RJ     AUV         ADJUST THE *USE* VECTOR               GPO          2441          I
          -FCC0509
GPO                MX6    59                                                FCC0509         9     A
GPO                RJ     AUV         MERGE, SQZ BLOCK AND ADJUST BIT VECTORS FCC0509        10     A
GPO                                                                         GPO          2442     A
GPO       *        REWRITE THE HOLDING BLOCK AS AN EMPTY BLOCK ( BOS, EOQ ) GPO          2443     A
GPO                                                                         GPO          2444     A
GPO                SA1    O.TXT                                             GPO          2445     A
GPO                SA2    O.MOD                                             GPO          2446     A
GPO                SB6    B0          TI = 0                                GPO          2447     A
GPO                SA5    X2          R1 = [[O.MOD]]                        GPO          2448     A
GPO                SB7    X1          TB = [O.TXT]                          GPO          2449     A
GPO                UX6    B2,X5                                             GPO          2450     A
GPO                SX7    8                                                 GPO          2451     A
GPO                RJ     SRI         SRI(  BOS  [R1] )                     GPO          2452     A
GPO                SB2    OC.EOQ                                            GPO          2453     A
GPO                SX6    0                                                 GPO          2454     A
GPO                RJ     SRI         SRI(  EOQ  0 )                        GPO          2455     A
GPO                SA4    BSIS                                              GPO          2456     A
GPO                SX7    B6                                                GPO          2457     A
GPO                SX6    X4          BSI = SETX( BSIS )                    GPO          2458     A
GPO                SA6    BSI                                               GPO          2459     A
GPO                SA7    L.TXT       L.TXT = 8                             GPO          2460     A
GPO                CALL   WMB         WRITE THE BLOCK BACK TO *BLK*         GPO          2461          I
          -CCGNC
GPO                CALL   WMB#        WRITE THE BLOCK BACK TO *BLK*         CCGNC         632     A
GPO                SX6    4                                                 GPO          2462     A
GPO                SA6    L.TXT       L.TXT = 4                             GPO          2463     A
GPO                EQ     CHB                                               GPO          2464     A
GPO        RDD     TITLE  RDD - REMOVE DEAD DEFINITIONS FROM A BLOCK        GPO          2465     A
GPO       **       RDD - REMOVE DEAD DEFINITIONS ( STORES ) FROM A BLOCK    GPO          2466     A
GPO       *                                                                 GPO          2467     A
GPO       *        ENTRY  *PCC* REGISTER CALL CONVENTIONS                   GPO          2468     A
GPO       *                                                                 GPO          2469     A
GPO       *        SCAN BLOCK FOR DEAD STORES ( ^(LX ! (DBU & USE ) )  ).   GPO          2470     A
GPO       *        CHANGE THEM TO *NOP*"S, CALL *SQZB* IF BLOCK CONTAINS DEAD GPO        2471     A
GPO       *        COMPUTATIONS ( VERY UNLIKELY ).                          GPO          2472     A
GPO       *        CALLED PRIOR TO LEVEL 2 / F.P. EXPANSION                 GPO          2473     A
GPO                                                                         GPO          2474     A
GPO        RDDA    CON    0           HEADER FOR LD LINKS                   FCCA499X        5     A
GPO                                                                         FCCA499X        6     A
GPO        RDD     ROUTINE                                                  GPO          2475     A
GPO                                                                         GPO          2476     A
GPO                RJ     IPS         INSERT POST STORES                    GPO          2477     A
GPO                ZR     X6,RDD0     IF NPS = 0                            GPO          2478     A
GPO                                                                         GPO          2479     A
GPO                SA4    BSW                                               GPO          2480     A
GPO                MX0    -R1.RIL+1                                         GPO          2481          I
          -CCG11
GPO                LX4    -IM.BIP                                           GPO          2482          I
          -CCG11

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                SA5    O.SEQ                                                GPO        2483     A
GPO                LX0    1                                                    GPO        2484          I
         -CCG11
GPO                SX0    RN.MASK                                              CCG11       116     A
GPO                LX4    -IM.BIP                                              CCG11       117     A
GPO                FBVA   X4          BVA = O.BVT + BVI[BIT(BN)]               GPO        2485     A
GPO                                                                            GPO        2486     A
GPO       RDD0     S"TB"  X5+2        TB = O.SEQ + 2                           GPO        2487     A
GPO                                                                            GPO        2488     A
GPO      *         SV = LX ! ( DBU & USE ) , LIVE ON EXIT OR USED IN BLOCK    GPO        2489     A
GPO      *         AND DDB = ^SV & DEF , DEAD DEFINITIONS.                    GPO        2490     A
GPO                                                                            GPO        2491     A
GPO                SA1    X7          DA = BVA                                 GPO        2492     A
GPO                SB6    B2+B2                                                GPO        2493     A
GPO                SA2    X7+B2       UFA = BVA + VL                           GPO        2494     A
GPO                SA3    X7+B6       UA = BVA + 2*VL                          GPO        2495     A
GPO                SA4    A3+B2       LXA = BVA + 3*VL                         GPO        2496     A
GPO                SA5    SVA                                                  GPO        2497     A
GPO                BX2    -X2*X1      DBU = DEF & ^UBD                         GPO        2498     A
GPO                SB5    X5                      (B5) = SVA                   GPO        2499     A
GPO                BX2    X2*X3                                                GPO        2500     A
GPO                BX6    X4+X2       SV = LX ! ( DBU & USE )                  GPO        2501     A
GPO                SA6    B5                                                   GPO        2502     A
GPO                BX7    -X6*X1      DDB = ^SV & DEF  */ DEAD DEF BITS        GPO        2503     A
GPO                EQ     B2,B1,RDD2  IF VL = 1                                GPO        2504     A
GPO       RDD1     SA1    A1+B1                                                GPO        2505     A
GPO                SA2    A2+B1                                                GPO        2506     A
GPO                SA3    A3+B1                                                GPO        2507     A
GPO                SA4    A4+B1                                                GPO        2508     A
GPO                BX2    -X2*X1                                               GPO        2509     A
GPO                SB2    B2-B1                                                GPO        2510     A
GPO                BX2    X2*X3                                                GPO        2511     A
GPO                BX6    X4+X2                                                GPO        2512     A
GPO                BX5    -X6*X1                                               GPO        2513     A
GPO                SA6    A6+B1                                                GPO        2514     A
GPO                BX7    X5+X7                                                GPO        2515     A
GPO                GT     B2,B1,RDD1                                           GPO        2516     A
GPO                                                                            GPO        2517     A
GPO       RDD2     CX4    X7                                                   GPO        2518     A
GPO                ZR     X4,RDD      IF COUNT( DDB ) = 0  */ NO DEAD DEFS IN BLO  GPO      2519     A
GPO                                                                            GPO        2520     A
GPO                PRNTABV  RDD,SVA                                            GPO        2521     A
GPO       RDD      SPACE  3,10                                                 GPO        2522     A
GPO      *         NOW SCAN BACKWARDS FOR DEAD STORES, ADJUST THE *PS* BIT    GPO        2523     A
GPO                                                                            GPO        2524     A
GPO                SA3    "TB"-B1     R2 = TB - 1                              GPO        2525     A
GPO                SA0    B0          DC = 0        */ DEAD CODE FLAG          GPO        2526     A
GPO                SA4    O.UDT                                                GPO        2527     A
GPO                SB6    X4+B1       UB = O.UDT + 1                           GPO        2528     A
GPO                LX3    -R2.TXTLP                                            GPO        2529     A
GPO                SB2    X3-4        L = TXTL[R2] - 4                         GPO        2530     A
GPO                SA4    "TB"+B2     DI = TB + L                              GPO        2531     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                MX1    -D.USESL                                                GPO          2532     A
GPO                LX1    D.USESP                                                 GPO          2533     A
GPO                SB3    59-D.^DP                                                GPO          2534     A
GPO                SB4    4                                                       GPO          2535     A
GPO                EQ     RDD4                                                    GPO          2536     A
GPO                                                                               GPO          2537     A
GPO       *        INDICATE THAT R-NUM *PS* FOR A LIVE *ST*                        GPO          2538     A
GPO                                                                               GPO          2539     A
GPO        RDD3    LX4    59-D.L2P                                                GPO          2540     A
GPO                MI     X4,RDD4     IF L2[DI]     */ LEVEL 2 *ST*               GPO          2541          I
          -CCG0208
GPO                SB2    X2+B1                                                   GPO          2542          I
          -CCG0208
GPO                PL     X4,RDD3A    IF ^L2[DI]        */ NOT A LEVEL 2 *ST*     CCG0208         5     A
GPO                LX4    D.L2P-D.XMTP                                            CCG0208         6     A
GPO                PL     X4,RDD4     IF ^XMT[DI]      */ NOT A LEVEL 0 *ST*      CCG0208         7     A
GPO                                                                               CCG0208         8     A
GPO        RDD3A   SB2    X2+B1                                                   CCG0208         9     A
GPO                SX6    B1                                                      GPO          2543     A
GPO                SA6    "TB"+B2     [TB+R+1] = 1                                GPO          2544     A
GPO                                                                               GPO          2545     A
GPO        RDD4    SA4    A4-B4       DI = DI - 4                                 GPO          2546     A
GPO                LX6    B3,X4                                                   GPO          2547     A
GPO                MI     X6,RDD5     IF ^D[DI]                                   GPO          2548     A
GPO                LX6    D.^DP-D.PSP                                             GPO          2549     A
GPO                MI     X6,RDD4A    IF PS[DI]                                   GPO          2550     A
GPO                LX6    D.PSP-D.LDP                                             GPO          2551     A
GPO                PL     X6,RDD4     IF ^LD[DI]                                  GPO          2552     A
GPO                                                                               GPO          2553     A
GPO       *        LD - SET BIT IN *SVA* SO WE DONT KILL DEFS THAT COME BEFORE     GPO          2554     A
GPO       *        AN EXTERNAL REF   ( USE,DEF EXTREF USE  SITUATION ).            GPO          2555     A
GPO       *        FORM LINK LIST OF ALL LDS.  THIS WILL BE SCANNED TO             FCCA499X        7     A
GPO       *        DETERMINE IF ANY STORES INTERFERE WITH SUBSEQUENT LOADS.        FCCA499X        8     A
GPO                                                                               GPO          2556     A
GPO                SA5    A4-2        R1 = DI - 2                                 GPO          2557     A
GPO                LX5    -R1.INP                                                 GPO          2558     A
GPO                SA3    B6+X5       U2 = UDT(IN[R1]+1)                          GPO          2559     A
GPO                SX2    B1                                                      GPO          2560     A
GPO                SA5    B5+X3       SVW = SVA(WI[U2])                           GPO          2561     A
GPO                UX7    B2,X3       B = BITN[U2]                                GPO          2562     A
GPO                LX2    B2,X2                                                   GPO          2563     A
GPO                BX6    X2+X5       SVW = SVW ! SHIFT(1,B)                      GPO          2564     A
GPO                SA6    A5                                                      GPO          2565     A
GPO                SA3    RDDA                                                    FCCA499X        9     A
GPO                SX7    X3                                                      FCCA499X       10     A
GPO                SA7    A4+B1       LDLNK = DI + 1;[LDLNK] = RDDA               FCCA499X       11     A
GPO                SX6    A7                                                      FCCA499X       12     A
GPO                SA6    A3          RDDA = LDLNK                                FCCA499X       13     A
GPO                EQ     RDD4                                                    GPO          2566     A
GPO                                                                               GPO          2567     A
GPO       *        INSTRUCTION *PS* - CLEAR BIT IF ALL DEFS KILLED                 GPO          2568     A
GPO                                                                               GPO          2569     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO       RDD4A    SA5    A4+B1       LI = DI + 1;  PSF = [LI]                    GPO        2570    A
GPO                MX7    0                                                       GPO        2571    A
GPO                SA7    A5          [LI] = 0                                    GPO        2572    A
GPO                NZ     X5,RDD4     IF PSF " 0    */ STILL *PS*                 GPO        2573    A
GPO                SX5    B1                                                      GPO        2574    A
GPO                LX5    D.PSP                                                   GPO        2575    A
GPO                BX6    -X5*X4      PS[DI] = 0                                  GPO        2576    A
GPO                SA6    A4                                                      GPO        2577    A
GPO                EQ     RDD4                                                    GPO        2578    A
GPO                                                                               GPO        2579    A
GPO       *        INSTRUCTION ^D                                                GPO        2580    A
GPO                                                                               GPO        2581    A
GPO       RDD5     SA5    A4-2        R1 = DI - 2                                 GPO        2582    A
GPO                UX7    B2,X5                                                   GPO        2583    A
GPO                LX6    D.^DP-D.STP                                             GPO        2584    A
GPO                EQ     B2,B1,RDD8  IF OC[R1] = OC.BOS                          GPO        2585         I
          -FCCA499X
GPO                EQ     B2,B1,RDD50 IF OC[R1] = OC.BOS                          FCCA499X     14    A
GPO                MI     X6,RDD6     IF ST[DI]                                   GPO        2586    A
GPO                SB2    B2-OC.RJ3                                               GPO        2587    A
GPO                MI     B2,RDD4     IF OC[R1] < OC.RJ3                          GPO        2588    A
GPO                GT     B2,B1,RDD4  IF OC[R1] > OC.RJ6                          GPO        2589    A
GPO                ERRNZ  OC.RJ3+1-OC.RJ6                                         GPO        2590    A
GPO                                                                               GPO        2591    A
GPO                SA3    A5-4        R1P = R1 - 4                                GPO        2592         I
          -FCC0612
GPO                SA3    A5-8        R1P = R1 - 8                                FCC0612       5    A
GPO                SA5    A5+B1       R2 = R1 + 1                                 GPO        2593    A
GPO                UX6    B2,X3                                                   GPO        2594    A
GPO                SX7    B2-OC.JIN                                               GPO        2595    A
GPO                ZR     X7,RDD4     IF OC[R1P] = OC.JIN  */ COMPUTED GOTO       GPO        2596    A
GPO                IF     DEF,.FTN,2
>>>>      CCGA035 20                                                              CCGA035      14    A
GPO                SA3    =XHO$TB
>>>>      CCGA035 21                                                              CCGA035      15    A
GPO                MI     X3,RDD50    IF TRACEBACK SELECTED
>>>>      CCGA035 22                                                              CCGA035      16    A
GPO                MX7    -FI.FTL                                                 GPO        2597    A
GPO                LX5    -FI.FTP                                                 GPO        2598    A
GPO                BX6    -X7*X5      F = FT[R2]    FUNCTION TYPE                 GPO        2599    A
GPO                SB2    X5-2                                                    GPO        2600         I
          -FCC0612
GPO                SB2    X6-2                                                    FCC0612       6    A
GPO                ZR     B2,RDD4     IF F = 2      */ *BEF*                      GPO        2601    A
GPO                                                                               GPO        2602    A
GPO                EQ     RDD8        QUIT UNTIL WE ADD MORE CODE                 GPO        2603         I
          -FCCA499X
GPO                EQ     RDD50       QUIT UNTIL WE ADD MORE CORE                 FCCA499X     15    A
GPO                                                                               GPO        2604    A
GPO       RDD6     BX2    "RN"X5      R = RI[R1]                                  GPO        2605    A
GPO                LX5    -R1.INP                                                 GPO        2606    A
GPO                SB2    X5          I = IN[R1]                                  GPO        2607    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                SA5    B6+X5       U2 = UDT(I+1)                                GPO          2608     A
GPO                ZR     B2,RDD3     IF I = 0      */ NO *UDT* INDEX              GPO          2609     A
GPO                LX6    D.STP-D.RFP                                              GPO          2610          I
          -FCC1556X
GPO                MI     X6,RDD3     IF RF[DI]     */ MOVED STORE                 GPO          2611          I
          -FCC1556X
GPO                                                                                GPO          2612     A
GPO       *        SEE IF STORE IS LIVE                                            GPO          2613     A
GPO                                                                                GPO          2614     A
GPO                SA3    B5+X5       SW = [SVA+WI[U2]]                            GPO          2615     A
GPO                UX7    B2,X5                                                    GPO          2616     A
GPO                SB2    B2-59       B = 59-BITN[U2]                              GPO          2617     A
GPO                SA5    A4-B1       R2 = DI - 1                                  GPO          2618     A
GPO                LX6    -B2,X3                                                   GPO          2619     A
GPO                MI     X6,RDD3     IF SHIFT(SW,B) < 0  */ STORE IS LIVE         GPO          2620     A
GPO                LX5    -IH.RFP                                                  GPO          2621     A
GPO                SB2    X5                                                       GPO          2622     A
GPO                GT     B2,B1,RDD3  IF RF[R2] > 1                                GPO          2623     A
GPO                BX6    X5          R2ST = R2                                    FCCA499X       16     A
GPO                SA3    RDDA        LDLNK = RDDA                                 FCCA499X       17     A
GPO                LX6    IH.RFP-IH.IHP                                            FCCA499X       18     A
GPO                                                                                FCCA499X       19     A
GPO       *        SEARCH THRU LINK LIST FOR INTERFERENCE BETWEEN THIS STORE       FCCA499X       20     A
GPO       *        AND SUBSEQUENT LOADS.                                           FCCA499X       21     A
GPO                                                                                FCCA499X       22     A
GPO        RDD10   ZR     X3,RDD30    IF LDLNK = 0   */END OF LIST                 FCCA499X       23     A
GPO                SA5    X3-2        R2LD = LDLNK - 2                             FCCA499X       24     A
GPO                LX5    -IH.IHP                                                  FCCA499X       25     A
GPO                BX7    X5-X6                                                    FCCA499X       26     A
GPO                LX6    IH.IHP-IH.CAP                                            FCCA499X       27     A
GPO                SX7    X7                                                       FCCA499X       28     A
GPO                NZ     X7,RDD20    IF IH[R2ST] " IH[R2LD]                       FCCA499X       29     A
GPO                LX5    IH.IHP-IH.RFP                                            FCCA499X       30     A
GPO                SX7    X5                                                       FCCA499X       31     A
GPO                NZ     X7,RDD3     IF RF[R2LD] " 0                              FCCA499X       32     A
GPO                LX5    IH.RFP-IH.CAP                                            FCCA499X       33     A
GPO                BX7    X5-X6                                                    FCCA499X       34     A
GPO                SX7    X7                                                       FCCA499X       35     A
GPO                ZR     X7,RDD3     IF CA[R2LD] = CA[R2ST]                       FCCA499X       36     A
GPO        RDD20   SA3    X3          LDLNK = [LDLNK]                              FCCA499X       37     A
GPO                LX6    IH.CAP-IH.IHP                                            FCCA499X       38     A
GPO                EQ     RDD10                                                    FCCA499X       39     A
GPO                                                                                GPO          2624     A
GPO       *        *ST* IS DEAD, CHANGE IT TO A *NOP* AND ADJUST USES OF PRED.     GPO          2625     A
GPO                                                                                GPO          2626     A
GPO                SA3    "TB"+X2     DP = TB + R                                  GPO          2627          I
          -FCCA499X
GPO        RDD30   SA3    "TB"+X2     DP = TB + R                                  FCCA499X       40     A
GPO                SX4    B1                                                       GPO          2628     A
GPO                IX6    X3-X4       USES[DP] = USES[DP] - 1                      GPO          2629     A
GPO                SA6    A3                                                       GPO          2630     A
GPO                BX7    -X1*X6                                                   GPO          2631     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                SA0    A0+B1       DC = DC + 1   */ INC DEAD STORE COUNT          GPO         2632     A
GPO                NZ     X7,RDD7     IF USES[DP] " 0                                GPO         2633          I
          -FCC0625
GPO                LX3    59-D.RFP                                                   FCC0625        5     A
GPO                AX3    59                                                         FCC0625        6     A
GPO                BX7    -X3*X7                                                     FCC0625        7     A
GPO                NZ     X7,RDD7     IF USES[DP] " 0 & ^RF[DP]                      FCC0625        8          I
          -FCCA499X
GPO                NZ     X7,RDD40    IF USES[DP] " 0 & ^RF[DP]                      FCCA499X      41     A
GPO                SA0    1S17        DC = -377777B */ INDICATE DEAD COMPUTATIONS    GPO         2634     A
GPO                                                                                  GPO         2635     A
GPO       RDD7     SA5    F.RDT+OC.NOP                                               GPO         2636          I
          -FCCA499X
GPO       RDD40    SA5    F.RDT+OC.NOP                                               FCCA499X      42     A
GPO                SB2    OC.NOP                                                     GPO         2637     A
GPO                PX6    B2,X2                                                      GPO         2638     A
GPO                SA6    A4-2        [DI-2] = TYI(OC.NOP,0,0,R)                     GPO         2639     A
GPO                BX7    X5                                                         GPO         2640     A
GPO                SA7    A4          [DI] = RDT(OC.NOP)                             GPO         2641     A
GPO                EQ     RDD4                                                       GPO         2642     A
GPO                                                                                  GPO         2643     A
GPO       RDD8     SX1    A0                                                         GPO         2644          I
          -FCCA499X
GPO       RDD50    SX1    A0                                                         FCCA499X      43     A
GPO                TRACE  RDD,RLIST,SEQ                                              GPO         2645     A
GPO                                                                                  GPO         2646     A
GPO                ZR     X1,RDD      IF DC = 0     */ NO STORES KILLED              GPO         2647          I
          -FCCA499X
GPO                PL     X1,RDD9     IF DC > 0     */ DEAD STORES ONLY              GPO         2648          I
          -FCCA499X
GPO       *        CLEAR LOAD LINKS.                                                 FCCA499X      44     A
GPO                                                                                  FCCA499X      45     A
GPO                SA2    RDDA        LDLNK = RDDA                                   FCCA499X      46     A
GPO                MX7    0                                                          FCCA499X      47     A
GPO       RDD54    ZR     X2,RDD56    IF LDLNK = 0   */END OF LIST                   FCCA499X      48     A
GPO                SA7    A2          [LDLNK] = 0                                    FCCA499X      49     A
GPO                SA2    X2          LDLNK = OLD [LDLNK]                            FCCA499X      50     A
GPO                EQ     RDD54                                                      FCCA499X      51     A
GPO                                                                                  FCCA499X      52     A
GPO       RDD56    ZR     X1,RDD      IF DC = 0   */NO DEAD STORES                   FCCA499X      53     A
GPO                PL     X1,RDD60    IF DC > 0   */DEAD STORES ONLY                 FCCA499X      54     A
GPO                                                                                  GPO         2649     A
GPO                MX6    1           L.MOD = 0                                      GPO         2650     A
GPO                CALL   MPB         RE SQUEEZE AND REWRITE THE BLOCK               GPO         2651     A
GPO                EQ     RDD                                                        GPO         2652     A
GPO                                                                                  GPO         2653     A
GPO                                                                                  GPO         2654     A
GPO       RDD9     MX6    0                                                          GPO         2655          I
          -FCCA499X
GPO       RDD60    MX6    0                                                          FCCA499X      55     A
GPO                CALL   MPB         MOVE THE BLOCK TO *TXT*                        GPO         2656     A
GPO                CALL   CUC         RESET THE USES COUNTS                          GPO         2657          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

-CCGNC

```
GPO                CALL   CUC#        RESET THE USES COUNTS                          CCGNC        633     A
GPO                CALL   WTB         REWRITE THE BLOCK                              GPO         2658     A
GPO                EQ     RDD                                                       GPO         2659     A
GPO       IPS      TITLE  IPS - INSERT POST STORES                                  GPO         2660     A
GPO      **        IPS - INSERT POST STORES INTO A BLOCK                            GPO         2661     A
GPO      *                                                                          GPO         2662     A
GPO      *         ENTRY  (X4) = [BSW]                                              GPO         2663     A
GPO      *                                                                          GPO         2664     A
GPO      *         EXIT   (X6) = NPS = 0 IF NO POST STORES INTRODUCED               GPO         2665     A
GPO                                                                                 GPO         2666     A
GPO       IPS      ROUTINE                                                          GPO         2667     A
GPO                LX4    -IM.BIP                                                   GPO         2668     A
GPO                FBVA   X4          BIW = BIT(BN);  BVA = O.BIT + BVI[BIW]        GPO         2669     A
GPO                LX4    -BI.PIIP                                                  GPO         2670     A
GPO                MX3    -BI.PIIL                                                  GPO         2671     A
GPO                BX6    -X3*X4      PSI = PII[BIW]  */ INDEX TO POST ST INFO      GPO         2672     A
GPO                ZR     X6,IPS      IF PSI = 0    */ NO POST STORES               GPO         2673     A
GPO                                                                                 GPO         2674     A
GPO      *         SCAN POST STORE LIST AND ADJUST *DEF* AND *UBD* BIT VECTORS      GPO         2675     A
GPO                                                                                 GPO         2676     A
GPO                SA1    O.PSI                                                     GPO         2677     A
GPO                SA2    O.UDT                                                     GPO         2678     A
GPO                IX3    X1+X6                                                     GPO         2679     A
GPO                SA7    VI2                                                       GPO         2680     A
GPO                SB7    X7          BVA = BVA                                     GPO         2681     A
GPO                SA5    X3-1        PI = O.PSI+PSI-1  */ HEADER WORD              GPO         2682     A
GPO                SB6    X2+B1       UB = O.UDT + 1                                GPO         2683     A
GPO                SB4    X5          NPS = [PI]    */ N. POST STORES               GPO         2684     A
GPO                SB5    X5          N = NPS                                       GPO         2685     A
GPO                PX0    B4,X6                                                     GPO         2686     A
GPO                SX1    B1                                                        GPO         2687     A
GPO                                                                                 GPO         2688     A
GPO       IPS1     SA5    A5+B1       PI = PI + 1                                   GPO         2689     A
GPO                SB5    B5-B1       N = N - 1                                     GPO         2690     A
GPO                LX5    -PS.UDIP                                                  GPO         2691     A
GPO                SA4    B6+X5       U2 = UDT( UDI[PI]+1 )                         GPO         2692     A
GPO                UX7    B3,X4       B = BITN[U2];  W = WI[U2]                     GPO         2693     A
GPO                SA2    B7+X4       BDA = BVA + W */ WORD OF DEF VECTOR           GPO         2694     A
GPO                LX4    B3,X1                                                     GPO         2695     A
GPO                SA3    A2+B2       UFA = BDA + VL */ UBD WORD                    GPO         2696     A
GPO                BX6    X4+X2       [BDA] = [BDA] ! SHIFT(1,B)                    GPO         2697     A
GPO                SA6    A2                                                        GPO         2698     A
GPO                BX7    -X4*X3      [UFA] = ^SHIFT(1,B) & [UFA]                   GPO         2699     A
GPO                SA7    A3                                                        GPO         2700     A
GPO                NZ     B5,IPS1     IF N " 0                                      GPO         2701     A
GPO                                                                                 GPO         2702     A
GPO      *         RESCAN LIST AND GENERATE A DEF, SA/XMT , ST FOR EACH ENTRY.      GPO         2703     A
GPO                                                                                 GPO         2704     A
GPO                SA1    O.SEQ                                                     GPO         2705     A
GPO                SA2    X1+4        R1 = [O.SEQ+4]  */ FIRST AFTER *BOS*          GPO         2706     A
GPO                SX1    4                                                         GPO         2707     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                UX6    B2,X2                                                GPO        2708    A
GPO                MX3    0            II = 0                                  GPO        2709    A
GPO                SX7    B2-OC.LAB                                            GPO        2710    A
GPO                NZ     X7,IPS2      IF OC[R1] " OC.LAB                      GPO        2711    A
GPO                BX3    X1           II = 4                                  GPO        2712    A
GPO        IPS2    LX3    ML.IIP                                               GPO        2713    A
GPO                LX1    ML.MTIP                                              GPO        2714    A
GPO                SX2    B4+B4                                                GPO        2715    A
GPO                BX1    X1+X3                                                GPO        2716    A
GPO                SX5    X2+B4                                                GPO        2717    A
GPO                LX5    ML.NIP                                               GPO        2718    A
GPO                BX1    X5+X1                                                GPO        2719    A
GPO                ADDWRD MLT,X1       ADDWRD( MLT , MCW(0,II,3*NPS,4) )       GPO        2720    A
GPO                LX5    -ML.NIP+2                                            GPO        2721    A
GPO                ALLOC  MOD,X5       ALLOC( MOD , 12*NPS )                   GPO        2722    A
GPO                SA4    O.PSI                                                GPO        2723    A
GPO                SB7    X2           TB = O.MOD                              GPO        2724    A
GPO                UX6    B3,X0        N = NPS                                 GPO        2725    A
GPO                IX7    X6+X4                                                GPO        2726    A
GPO                SA5    X7           PI = O.PSI + IPSA                       GPO        2727    A
GPO                SB4    B3+B3                                                GPO        2728    A
GPO                SB5    B6           MI = OLD(L.MOD)  */ INDEX FOR DEFS      GPO        2729    A
GPO                SB4    B4+B4                                                GPO        2730        I
          -FCC0408
GPO                SA1    O.SYM                                                FCC0408      73        I
          -FCC0656
GPO                SB4    B4+B4                                                FCC0408      74        I
          -FCC0656
GPO                SA0    X1                        (A0) = O.SYM               FCC0408      75        I
          -FCC0656
GPO                SB4    B4+B4                     (B4) = 4*NPS               FCC0656      41    A
GPO                                                                           GPO        2731    A
GPO        IPS3    MX1    PS.REGL                                              GPO        2732    A
GPO                LX1    PS.REGL+PS.REGP                                      GPO        2733    A
GPO                BX2    X1*X5                                                GPO        2734    A
GPO                LX2    R1.SOP-PS.REGP                                       GPO        2735    A
GPO                SX1    B5                                                   GPO        2736    A
GPO                BX6    X2+X1                                                GPO        2737    A
GPO                SB2    OC.DEF                                               GPO        2738    A
GPO                MX7    0                                                    GPO        2739    A
GPO                SB6    B5           TI = MI                                 GPO        2740    A
GPO                LX2    -R1.SOP                                              GPO        2741        I
          -FCC1025
GPO                LX2    58-R1.SOP-SO.RTP                                     FCC1025       7    A
GPO                RJ     SRI          SRI( DEF  MI,REG[PI] )                  GPO        2742    A
GPO                SB6    B5+B4        TI = MI + 4*NPS                         GPO        2743    A
GPO                SX6    B6                                                   GPO        2744    A
GPO                LX1    R1.RJP                                               GPO        2745    A
GPO                BX6    X1+X6                                                GPO        2746    A
GPO                SB2    OC.SA                                                GPO        2747        I
          -FCC1025
GPO                SX2    X2-20B                                               GPO        2748        I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
          -FCC1025
GPO                MI     X2,IPS4     IF REG[PI] < 20B  */ IN A B/A REG          GPO        2749        I
          -FCC1025
GPO                SB2    OC.XMT                                                 GPO        2750        I
          -FCC1025
GPO        IPS4    RJ     SRI         SRI( SA/XMT MI+4*NPS,MI )                  GPO        2751        I
          -FCC1025
GPO                SB2    OC.XMT      OC = OC.XMT;  R2W = 0                      FCC1025       8     A
GPO                MI     X2,IPS4     IF REG[PI] \ 20B                           FCC1025       9     A
GPO                SB2    OC.SA       OC = OC.SA                                 FCC1025      10     A
GPO                MX1    -PS.CAIHL                                              FCC1025      11     A
GPO                BX7    -X1*X5      R2W = IHW(0,CAIH[PI])                      FCC1025      12     A
GPO        IPS4    RJ     SRI         SRI( OC  MI+4*NPS,MI, R2W )                FCC1025      13     A
GPO                SX6    B6-4        R = TI - 4                                 GPO        2752     A
GPO                SB6    X6+B4       TI = R + 4*NPS                             GPO        2753     A
GPO                MX1    -PS.CAIHL                                              GPO        2754     A
GPO                BX7    -X1*X5                                                 GPO        2755     A
GPO                LX5    -PS.UDIP                                               GPO        2756     A
GPO                SB2    OC.ST                                                  GPO        2757     A
GPO                SX5    X5                                                     GPO        2758     A
GPO                LX5    R1.INP                                                 GPO        2759     A
GPO                SB3    B3-B1       N = N - 1                                  GPO        2760     A
GPO                BX6    X5+X6                                                  GPO        2761     A
GPO                SB5    B5+4        MI = MI + 4                                GPO        2762        I
          -FCC0408
GPO                RJ     SRI         SRI( ST  R,0,CAIH[PI],UDI[PI] )            GPO        2763        I
          -FCC0408
GPO                LX1    B1,X7                                                  FCC0408      76        I
          -FCC0656
GPO                RJ     SRI         SRI( ST  R,0,CAIH[PI],UDI[PI] )            FCC0408      77        I
          -FCC0656
GPO                SB2    X1+B1                                                  FCC0408      78        I
          -FCC0656
GPO                SA1    A0-B2       WORDB = SYM(2*IH[PI]-1)                    FCC0408      79        I
          -FCC0656
GPO                LX1    -P.FPB                                                 FCC0408      80        I
          -FCC0656
GPO                SX2    B1          DI = MOD(MI+2)                             FCC0408      81        I
          -FCC0656
GPO                BX5    X2*X1                                                  FCC0408      82        I
          -FCC0656
GPO                LX5    D.FPP                                                  FCC0408      83        I
          -FCC0656
GPO                BX6    X5+X6       FP[DI] = FP[WORDB]                         FCC0408      84        I
          -FCC0656
GPO                SA6    A6                                                     FCC0408      85        I
          -FCC0656
GPO                SB5    B5+4        MI = MI + 4                                FCC0408      86        I
          -FCC0656
GPO                SB5    B5+4        MI = MI + 4                                FCC0656      42     A
GPO                RJ     SMI         SMI( ST  R,,CAIH[PI],UDI[PI] )             FCC0656      43     A
GPO                SA5    A5+B1       PI = PI + 1                                GPO        2764     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                NZ     B3,IPS3     IF N " 0                                  GPO        2765     A
GPO                                                                             GPO        2766     A
GPO                MX6    1                                                     GPO        2767          I
          -FCC0509
GPO                CALL   MPB         MERGE MODS AND REWRITE THE BLOCK          GPO        2768          I
          -FCC0509
GPO                SA5    VI2                                                   GPO        2769          I
          -FCC0509
GPO                RJ     AUV         ADJUST THE USE VECTOR                     GPO        2770          I
          -FCC0509
GPO                SA5    VI2                                                   FCC0509      11     A
GPO                MX6    1                                                     FCC0509      12     A
GPO                RJ     AUV         MERGE, SQZ BLOCK AND ADJUST BIT VECTORS   FCC0509      13     A
GPO                CALL   RTB         AND GET IT BACK                           GPO        2771          I
          -CCGNC
GPO                CALL   RTB#        AND GET IT BACK                           CCGNC       634     A
GPO                SX6    B1                                                    GPO        2772     A
GPO                EQ     IPS                                                   GPO        2773     A
GPO        AUV     TITLE  AUV - ADJUST USE VECTOR                               GPO        2774     A
GPO       **       AUV - ADJUST *USE* BIT VECTOR AFTER A *SQZB* CALL TO ACCOUNT GPO        2775     A
GPO       *               FOR ANY STORE/LOAD SQUEEZING THAT TOOK PLACE.         GPO        2776     A
GPO       *        THIS IS DONE PRIOR TO THE *RDD* CALL SO AS TO GET RID OF     GPO        2777     A
GPO       *        THE INTRA BLOCK USES.                                        GPO        2778     A
GPO       *                                                                     GPO        2779     A
GPO       *        ENTRY  (X5) = BVA , BLOCK BIT VECTOR ADDRESS                 GPO        2780     A
GPO       *                (X6) = *MPB* CALL FLAG                               FCC0509      14     A
GPO                                                                             GPO        2781     A
GPO        AUVA    BFMW   D,(LD,BM)                                            GPO        2782     A
GPO                                                                             GPO        2783     A
GPO        AUV     ROUTINE                                                      GPO        2784     A
GPO                SA1    SVA                                                   FCC0509      15     A
GPO                SA2    VL                                                    FCC0509      16     A
GPO                BX7    X5                                                    FCC0509      17     A
GPO                SA7    VI2                                                   FCC0509      18     A
GPO                IX7    X1+X2                                                 FCC0509      19     A
GPO                SA7    =XSUVA      SUVA = SVA+VL  */ SET FLAG FOR SQZ/SMR    FCC0509      20     A
GPO                LX2    1                                                     FCC0509      21     A
GPO                SB6    X1                                                    FCC0509      22     A
GPO                SB7    B6+X2                                                 FCC0509      23     A
GPO                MX7    0                                                     FCC0509      24     A
GPO        AUV1    SA7    B6          FOR I = 1 TO 2*VL;  SVA(I) = 0            FCC0509      25     A
GPO                SB6    B6+B1                                                 FCC0509      26     A
GPO                LT     B6,B7,AUV1                                            FCC0509      27     A
GPO                                                                             FCC0509      28     A
GPO                CALL   MPB         MERGE MODS AND SQZ BLOCK                  FCC0509      29     A
GPO                SX7    0                                                     FCC0509      30     A
GPO                SA7    SUVA        SUVA = 0                                  FCC0509      31     A
GPO                SA5    VI2                                                   FCC0509      32     A
GPO                SA1    =XSTLDS                                               GPO        2785     A
GPO                ZR     X1,AUV      IF STLDS = 0  */ NO STORE/LD SQUEEZING    GPO        2786     A
GPO                                                                             GPO        2787     A
GPO                SA2    SVA                                                   GPO        2788     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                SA3    VL                                                    GPO        2789     A
GPO                SA1    O.UDT                                                 GPO        2790     A
GPO                SA4    O.TXT                                                 GPO        2791     A
GPO                LX6    B1,X3                                                 GPO        2792     A
GPO                SB4    X3                        (B4) = VL                   GPO        2793     A
GPO                SB5    X2                        (B5) = SVA                  GPO        2794     A
GPO                SB7    X5+B4       UFA = BVA + VL  */ USE FIRST ADDRESS      GPO        2795     A
GPO                MX6    0                                                     GPO        2796          I
          -FCC0509
GPO                SA0    X1+B1                     (A0) = O.UDT + 1            GPO        2797          I
          -FCC0509
GPO                SB3    B0                                                    GPO        2798          I
          -FCC0509
GPO       AUV1     SA6    B5+B3       FOR I = 1 TO VL;  SVA(I) = 0              GPO        2799          I
          -FCC0509
GPO                SB3    B3+B1                                                 GPO        2800          I
          -FCC0509
GPO                LT     B3,B4,AUV1                                            GPO        2801          I
          -FCC0509
GPO                                                                             GPO        2802          I
          -FCC0509
GPO                SA0    X1+B1                     (A0) = O.UDT + 1            FCC0509      33     A
GPO                SA1    AUVA                                                  GPO        2803     A
GPO                SB2    B1+B1                                                 GPO        2804     A
GPO                SA4    X4+B2       DI = O.TXT + 2                            GPO        2805     A
GPO                SX7    B1                                                    GPO        2806     A
GPO                EQ     AUV3                                                  GPO        2807     A
GPO                                                                             GPO        2808     A
GPO       *        SET BIT FOR REFERENCED VARIABLE                               GPO        2809     A
GPO                                                                             GPO        2810     A
GPO       AUV2     LX5    -R1.INP                                               GPO        2811     A
GPO                SB3    X5          I = IN[R1]                                GPO        2812     A
GPO                ZR     B3,AUV3     IF I = 0                                  GPO        2813     A
GPO                                                                             GPO        2814     A
GPO                SA3    A0+B3       U2 = UDT(I+1);  W = WI[U2]                GPO        2815     A
GPO                SA2    B5+X3                                                 GPO        2816     A
GPO                UX5    B6,X3                                                 GPO        2817     A
GPO                LX4    B6,X7                                                 GPO        2818     A
GPO                BX6    X4+X2       SVA(W) = SVA(W) ! SHIFT(1,BITN[U2])       GPO        2819     A
GPO                SA6    A2                                                    GPO        2820     A
GPO                                                                             GPO        2821     A
GPO       *        ADVANCE TO NEXT INSTRUCTION                                   GPO        2822     A
GPO                                                                             GPO        2823     A
GPO       AUV3     SA5    A4+B2       R1 = DI + 2                               GPO        2824     A
GPO                SA4    A5+B2       DI = R1 + 2                               GPO        2825     A
GPO                BX6    X1*X4                                                 GPO        2826     A
GPO                LX4    59-D.LDP                                              GPO        2827     A
GPO                ZR     X6,AUV3     IF ^( LD[DI] ! BM[DI] )                   GPO        2828     A
GPO                MI     X4,AUV2     IF LD[DI]                                 GPO        2829     A
GPO                                                                             GPO        2830     A
GPO                UX6    B3,X5                                                 GPO        2831     A
GPO                ZR     B3,AUV4     IF OC[R1] = OC.EOQ  */ END OF SEQ         GPO        2832     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                SX6    B3-OC.RJ6                                          GPO        2833    A
GPO                NZ     X6,AUV3     IF OC[R1] " OC.RJ6                     GPO        2834    A
GPO                                                                           GPO        2835    A
GPO      *         USER OR I/O REFERENCE, EXIT SINCE WE HAVE LOST THE PARAMETER  GPO        2836    A
GPO      *         LIST INFORMATION AT THE END OF THE BLOCK DURING THE MERGE.  GPO        2837    A
GPO                                                                           GPO        2838    A
GPO                EQ     AUV                                                GPO        2839    A
GPO                                                                           GPO        2840    A
GPO      *         ADJUST *UBD* AND *USE* BIT VECTORS                         GPO        2841         I
          -FCC0509
GPO                                                                           GPO        2842         I
          -FCC0509
GPO       AUV4     SA2    B5+B3                                              GPO        2843         I
          -FCC0509
GPO                SA1    B7+B3                                              GPO        2844         I
          -FCC0509
GPO                BX6    X2          USE(I) = SVA(I)                        GPO        2845         I
          -FCC0509
GPO                SA6    A1+B4                                              GPO        2846         I
          -FCC0509
GPO                BX7    X2*X1       UBD(I) = UBD(I) & USE(I)               GPO        2847         I
          -FCC0509
GPO                SA7    A1                                                 GPO        2848         I
          -FCC0509
GPO                SB3    B3+1        I = I + 1                              GPO        2849         I
          -FCC0509
GPO      *         ADJUST *USE* AND *UBD* BV"S, USE = USE & ^(SVA2 & ^SVA)    FCC0509      34    A
GPO                                                                           FCC0509      35    A
GPO       AUV4     SA2    B5+B3                    SVA                       FCC0509      36    A
GPO                SA3    A2+B4                    SVA2                      FCC0509      37    A
GPO                SA1    B7+B3                    UBD                       FCC0509      38    A
GPO                SA4    A1+B4                    USE                       FCC0509      39    A
GPO                BX5    -X2*X3                                             FCC0509      40    A
GPO                BX6    -X5*X4      USE(I) = USE(I) & ^(SVA2(I) & ^SVA(I) )  FCC0509      41    A
GPO                SA6    A4                                                 FCC0509      42    A
GPO                BX7    X6*X1       UBD(I) = UBD(I) & USE(I)               FCC0509      43    A
GPO                SA7    A1                                                 FCC0509      44    A
GPO                SB3    B3+B1       I = I + 1                              FCC0509      45    A
GPO                LT     B3,B4,AUV4  IF I < VL                              GPO       2850    A
GPO                                                                           GPO       2851    A
GPO                PRNTABV  AUV,SVA                                          GPO       2852    A
GPO                EQ     AUV                                                GPO       2853    A
GPO       FEI      TITLE  FEI - FORM ENTRY INFORMATION                       FCCA513X     10    A
GPO      **        FEI - FORM ENTRY INFORMATION                              FCCA513X     11    A
GPO      *         COMPUTE REGION LIVE ENTRY BIT VECTOR.                     FCCA513X     12    A
GPO                                                                           FCCA513X     13    A
GPO       FEI      ROUTINE                                                   FCCA513X     14    A
GPO                SA4    HBI+1       BN = BN[HBI(2)]   */ HEADER NODE       FCCA513X     15    A
GPO                FBVA   X4          BVA = O.BVT + BVI[BIT(BN)]             FCCA513X     16    A
GPO                SA2    LEA                                                FCCA513X     17    A
GPO                SB3    B2                                                 FCCA513X     18    A
GPO                SB4    B2+B2                                              FCCA513X     19    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                SA3    X7          BDA = BVA                                  FCCA513X      20     A
GPO                SA4    X7+B2       UFA = BVA + VL  */ UBD                     FCCA513X      21     A
GPO                SA5    A4+B4       LXW = BVA + 3*VL  */ LX                    FCCA513X      22     A
GPO                BX0    -X3*X5                                                 FCCA513X      23     A
GPO                BX6    X0+X4       LEW = ^[BDA] & [LXW] ! [UFA]               FCCA513X      24     A
GPO                SA2    X2                                                     FCCA513X      25     A
GPO                BX6    X6+X2       [LEA] = LEW ! [LEA] */ ACCNT FOR MOVED ST"S FCCA513X      26     A
GPO                SA6    A2                                                     FCCA513X      27     A
GPO                SB2    B2-B1                                                  FCCA513X      28     A
GPO                ZR     B2,FEI      IF VL = 1                                  FCCA513X      29     A
GPO       FEI1     SA3    A3+B1       = BDA                                      FCCA513X      30     A
GPO                SA4    A4+B1       = UFA                                      FCCA513X      31     A
GPO                SA5    A5+B1       = LXW                                      FCCA513X      32     A
GPO                SB2    B2-B1       VL = VL - 1                                FCCA513X      33     A
GPO                SA2    A2+B1       = LEA                                      FCCA513X      34     A
GPO                BX0    -X3*X5                                                 FCCA513X      35     A
GPO                BX6    X0+X4                                                  FCCA513X      36     A
GPO                BX6    X6+X2                                                  FCCA513X      37     A
GPO                SA6    A2          LEA = ^[BDA] & [LXA] ! [UFA]               FCCA513X      38     A
GPO                NZ     B2,FEI1                                                FCCA513X      39     A
GPO                                                                              FCCA513X      40     A
GPO                EQ     FEI                                                    FCCA513X      41     A
GPO       FXI      TITLE  FXI - FORM EXIT INFORMATION                            GPO         2854     A
GPO      **        FXI - FORM EXIT INFORMATION                                   GPO         2855     A
GPO      *         COMPUTE REGION LIVE ENTRY BIT VECTOR.                         GPO         2856          I
         -FCCA513X
GPO      *         BUILD LIST OF SUCCESSORS OF *SCR* AND THEIR LIVE ENTRY VECTORS GPO         2857     A
GPO      *         FORM REGION LIVE EXIT AND MOVABLE DEF VECTORS.                GPO         2858     A
GPO                                                                              GPO         2859     A
GPO       FXI      ROUTINE                                                       GPO         2860     A
GPO                SA4    HBI+1       BN = BN[HBI(2)]   */ HEADER NODE           GPO         2861     A
GPO                FBVA   X4          BVA = O.BVT + BVI[BIT(BN)]                 GPO         2862     A
GPO                SA2    LEA                                                    GPO         2863          I
         -FCCA513X
GPO                SB3    B2                                                     GPO         2864          I
         -FCCA513X
GPO                SB4    B2+B2                                                  GPO         2865          I
         -FCCA513X
GPO                SA3    X7          BDA = BVA                                  GPO         2866          I
         -FCCA513X
GPO                SA4    X7+B2       UFA = BVA + VL  */ UBD                     GPO         2867          I
         -FCCA513X
GPO                SA5    A4+B4       LXW = BVA + 3*VL  */ LX                    GPO         2868          I
         -FCCA513X
GPO                BX0    -X3*X5                                                 GPO         2869          I
         -FCCA513X
GPO                BX6    X0+X4       LEW = ^[BDA] & [LXW] ! [UFA]               GPO         2870          I
         -FCCA513X
GPO                SA6    X2                                                     GPO         2871          I
         -FCC1556X -FCCA513X
GPO                SA2    X2                                                     FCC1556X       6          I
         -FCCA513X

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                 BX6    X6+X2       [LEA] = LEW ! [LEA] */ ACCNT FOR MOVED ST"S         FCC1556X       7          I
          -FCCA513X
GPO                 SA6    A2                                                              FCC1556X       8          I
          -FCCA513X
GPO                 SB2    B2-B1                                                           GPO         2872          I
          -FCCA513X
GPO                 ZR     B2,FXI2                                                         GPO         2873          I
          -FCCA513X
GPO        FXI1     SA3    A3+B1                                                           GPO         2874          I
          -FCCA513X
GPO                 SA4    A4+B1                                                           GPO         2875          I
          -FCCA513X
GPO                 SA5    A5+B1                                                           GPO         2876          I
          -FCCA513X
GPO                 SB2    B2-B1                                                           GPO         2877          I
          -FCCA513X
GPO                 SA2    A2+B1                                                           FCC1556X       9          I
          -FCCA513X
GPO                 BX0    -X3*X5                                                          GPO         2878          I
          -FCCA513X
GPO                 BX6    X0+X4                                                           GPO         2879          I
          -FCCA513X
GPO                 SA6    A6+B1                                                           GPO         2880          I
          -FCC1556X -FCCA513X
GPO                 BX6    X6+X2                                                           FCC1556X      10          I
          -FCCA513X
GPO                 SA6    A2                                                              FCC1556X      11          I
          -FCCA513X
GPO                 NZ     B2,FXI1                                                         GPO         2881          I
          -FCCA513X
GPO                 SB3    B0                                                              FCCA513X      42     A
GPO                                                                                        GPO         2882     A
GPO       *         INITIALIZE *MD* VECTOR AS *DEF(R)* ( MOVABLE DEFS )                     GPO         2883     A
GPO                                                                                        GPO         2884     A
GPO        FXI2     SA1    DVA                                                             GPO         2885          I
          -FCCA513X
GPO                 SA1    DVA                                                             FCCA513X      43     A
GPO                 SA2    MDA                                                             GPO         2886     A
GPO        FXI3     SA3    X1+B2       MOVE( VL , DVA , MDA )                              GPO         2887          I
          -FCCA513X
GPO        FXI2     SA3    X1+B3       MOVE( VL , DVA , MDA )                              FCCA513X      44     A
GPO                 BX6    X3                                                              GPO         2888     A
GPO                 SA6    X2+B2                                                           GPO         2889          I
          -FCCA513X
GPO                 SB2    B2+B1                                                           GPO         2890          I
          -FCCA513X
GPO                 LT     B2,B3,FXI3                                                      GPO         2891          I
          -FCCA513X
GPO                 SA6    X2+B3                                                           FCCA513X      45     A
GPO                 SB3    B3+B1                                                           FCCA513X      46     A
GPO                 LT     B3,B2,FXI2                                                      FCCA513X      47     A
GPO                                                                                        GPO         2892     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO      *         FORM *SCR* SUCCESSOR LIST. FIRST MARK THE NODES IN THE                GPO      2893     A
GPO      *         *SCR* AS *SCR* IN *BTT*.                                              GPO      2894     A
GPO                                                                                      GPO      2895     A
GPO                SA1    IBA                                                            GPO      2896     A
GPO                SA2    A1+B1                                                          GPO      2897     A
GPO                MX0    1                                                              GPO      2898     A
GPO                SA3    O.BTT                                                          GPO      2899     A
GPO                IX6    X1+X2                                                          GPO      2900     A
GPO                SB7    X3                                                             GPO      2901     A
GPO                SB6    X6          IPL = IBA + NN  */ LWA+1 OF INTERVAL LIST          GPO      2902     A
GPO                LX0    1+E.SCRP                                                       GPO      2903     A
GPO                SA5    B6          IP = IPL                                           GPO      2904     A
GPO                                                                                      GPO      2905     A
GPO       FXI4     SA5    A5-B1       IP = IP - 1                                        GPO      2906     A
GPO                LX5    59-IM.SCRP                                                     GPO      2907     A
GPO                PL     X5,FXI4     IF ^SCR[IP]                                        GPO      2908     A
GPO                                                                                      GPO      2909     A
GPO                LX5    1+IM.SCRP-IM.BNP                                               GPO      2910     A
GPO                SA4    B7+X5                                                          GPO      2911     A
GPO                BX6    X0+X4       SCR[BTT(BN[IP])] = 1                               GPO      2912     A
GPO                LX5    IM.BNP+59-IM.HNP                                               GPO      2913     A
GPO                SA6    A4+                                                            GPO      2914     A
GPO                PL     X5,FXI4     IF ^HN[IP]                                         GPO      2915     A
GPO                                                                                      GPO      2916     A
GPO      *         RESCAN THE INTERVAL LIST AND FORM THE SUCCESSOR LIST IN *BTT*         GPO      2917     A
GPO                                                                                      GPO      2918     A
GPO                SA1    EIT                                                            GPO      2919     A
GPO                SA2    O.BIT                                                          GPO      2920     A
GPO                MX6    0                                                              GPO      2921     A
GPO                SA6    B7          BTT(0) = 0                                         GPO      2922     A
GPO                SA0    B0          NX = 0        */ N. EXIT NODES                     GPO      2923     A
GPO                SB5    B0          LS = 0        */ LAST SUCCESSOR                    GPO      2924     A
GPO                SA5    B6          IP = IPL                                           GPO      2925     A
GPO                LX0    E.JPP-E.SCRP                                                   GPO      2926     A
GPO                SB6    X1          EIB = [EIT]   */ BASE OF EDGE INDEX TABLE          GPO      2927     A
GPO                SB4    X2                        (B4) = [O.BIT]                       GPO      2928     A
GPO                                                                                      GPO      2929     A
GPO       FXI5     SA5    A5-B1       IP = IP - 1                                        GPO      2930     A
GPO                LX5    59-IM.SCRP                                                     GPO      2931     A
GPO                PL     X5,FXI5     IF ^SCR[IP]                                        GPO      2932     A
GPO                LX5    1+IM.SCRP-IM.BNP                                               GPO      2933     A
GPO                                                                                      GPO      2934     A
GPO                SA4    B6+X5       EII = EIB + BN[IP]                                 GPO      2935     A
GPO                LX4    -EI.SUIP                                                       GPO      2936     A
GPO                UX6    B3,X4       N = NS[EII]                                        GPO      2937     A
GPO                SA3    B6+X4       SI = EIB + SUI[EII]                                GPO      2938     A
GPO                                                                                      GPO      2939     A
GPO      *         ADD SUCCESSORS OF NODE THAT ARE NOT *SCR* IN *BTT* TO EXIT LIS        GPO      2940     A
GPO                                                                                      GPO      2941     A
GPO       FXI6     LX3    -ET.SUCCP   J = SUCC[SI]  */ BN OF SUCCESSOR                   GPO      2942     A
GPO                SA2    B7+X3       B = BTT(J)                                         GPO      2943     A
GPO                SB3    B3-B1       N = N - 1                                          GPO      2944     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO              LX2    59-E.SCRP                                          GPO        2945    A
GPO              MI     X2,FXI8     IF SCR[B]     */ SUCC IS IN *SCR*      GPO        2946    A
GPO                                                                        GPO        2947    A
GPO              MX1    -E.NPREDL                                          GPO        2948    A
GPO              LX2    1+E.SCRP-E.NPREDP                                  GPO        2949    A
GPO              BX6    -X1*X2                                             GPO        2950    A
GPO              NZ     X6,FXI7     IF NPRED[B] " 0 */ NOT FIRST TIME REACHED GPO      2951    A
GPO                                                                        GPO        2952    A
GPO              SA4    B6+X3       EIJ = EIB + J                          GPO        2953    A
GPO              LX4    -EI.NPP                                            GPO        2954    A
GPO              SA0    A0+B1       NX = NX + 1                            GPO        2955    A
GPO              BX7    -X1*X4                                             GPO        2956    A
GPO              SX6    X5                                                 GPO        2957         I
          -FCC0985
GPO              SA4    B7+X5                                              FCC0985       7    A
GPO              SX6    X4                                                 FCC0985       8    A
GPO              BX2    X7+X2       NPRED[B] = NP[EIJ]  */ N.PRED OF SUCC  GPO        2958    A
GPO              LX6    E.PREDP-E.NPREDP                                   GPO        2959    A
GPO              SX7    B5                                                 GPO        2960    A
GPO              BX2    X6+X2       PRED[B] = BN[IP]  */ FIRST PRED TO REACH GPO      2961         I
          -FCC0985
GPO              BX2    X6+X2       PRED[B] = BTT(BN[IP])  */ FIRST PRED TO REACH FCC0985       9    A
GPO              LX7    E.LINKP-E.NPREDP                                   GPO        2962    A
GPO              BX2    X7+X2       LINK[B] = LS  */ POINT THIS TO LAST    GPO        2963    A
GPO              SB5    A2          LS = BTT(J)                            GPO        2964    A
GPO                                                                        GPO        2965    A
GPO       FXI7   SX1    B1                                                 GPO        2966    A
GPO              LX3    E.JPP-ET.JPP+ET.SUCCP                              GPO        2967    A
GPO              IX2    X2-X1       NPRED[B] = NPRED[B] - 1                GPO        2968    A
GPO              BX6    X0*X3                                              GPO        2969    A
GPO              LX2    E.NPREDP                                           GPO        2970    A
GPO              BX7    X6+X2       JP[B] = JP[B] ! JP[SI]                 GPO        2971    A
GPO              SA7    A2                                                 GPO        2972    A
GPO                                                                        GPO        2973    A
GPO       FXI8   SA3    A3+B1       SI = SI + 1                            GPO        2974    A
GPO              NZ     B3,FXI6     IF N " 0                               GPO        2975    A
GPO                                                                        GPO        2976    A
GPO              LX5    59-IM.HNP+IM.BNP                                   GPO        2977    A
GPO              PL     X5,FXI5     IF ^HN[IP]                             GPO        2978    A
GPO                                                                        GPO        2979    A
GPO              LX5    1+IM.HNP-IM.BIP                                    GPO        2980    A
GPO              SX6    X5                                                 GPO        2981    A
GPO              LX5    IM.BIP-IM.BNP                                      GPO        2982    A
GPO              SA6    B7+X5       BTT(BN[IP]) = BI[IP]  */ RESTORE HEADER ENT GPO   2983    A
GPO                                                                        GPO        2984    A
GPO              SX6    B5                                                 GPO        2985    A
GPO              SX7    A0                                                 GPO        2986    A
GPO              SA6    O.ENL                                              GPO        2987    A
GPO              SA7    N.ENL                                              GPO        2988    A
GPO                                                                        GPO        2989    A
GPO       *      SCAN THE EXIT NODE LIST AND MOVE THE LIST AND THE LIVE ENTRY GPO     2990    A
GPO       *      BIT VECTORS TO *RXI*.  FORMAT -                            GPO        2991    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO      *         0, BTT WORD,LE&DEFR,...,0                                    GPO        2992     A
GPO      *         COMPUTE THE REGION LIVE EXIT BIT VECTOR AND ADJUST *MD*.     GPO        2993     A
GPO                                                                             GPO        2994     A
GPO                SA5    BVL                                                   GPO        2995     A
GPO                SX3    X5+2                                                  GPO        2996     A
GPO                IX2    X7*X3                                                 GPO        2997     A
GPO                ALLOC  RXI,X2+2    ALLOC( RXI , NX*(BVL+2)+2 )               GPO        2998     A
GPO                MX7    0                                                     GPO        2999     A
GPO                SA7    X2          RI = [O.RXI];  [RI] = 0                   GPO        3000     A
GPO                SA1    O.BIT                                                 GPO        3001     A
GPO                SA2    LXA                                                   GPO        3002     A
GPO                SA3    MDA                                                   GPO        3003     A
GPO                SA4    O.ENL                                                 GPO        3004     A
GPO                SB4    X5+B1       VL = BVL + 1                              GPO        3005     A
GPO                SB5    B4+B4                                                 GPO        3006     A
GPO                SB6    X1                                                    GPO        3007     A
GPO                SB7    X2                                                    GPO        3008     A
GPO                SA0    X3                                                    GPO        3009     A
GPO                SA5    X4          XI = [O.ENL];  XW = [XI]                  GPO        3010     A
GPO                ZR     X4,FXI15    IF O.ENL = 0  */ LOOP WITHOUT EXITS       GPO        3011     A
GPO                                                                             GPO        3012     A
GPO       FXI10    MX4    -E.NPREDL                                             GPO        3013          I
          -FCC0392
GPO                SX0    B1          PSP = 1       */ POST STORE POSSIBLE      GPO        3014          I
          -FCC0392
GPO                LX4    E.NPREDP                                              GPO        3015          I
          -FCC0392
GPO                BX6    -X4*X5                                                GPO        3016          I
          -FCC0392
GPO                ZR     X6,FXI11    IF NPRED[XW] = 0  */ ALL PRED IN *SCR*    GPO        3017          I
          -FCC0392
GPO                BX7    X5                                                    GPO        3018          I
          -FCC0392
GPO      *         CHECK SUCCESSOR NODE, SEE IF POST STORES TO IT ARE POSSIBLE   FCC0392      89     A
GPO                                                                             FCC0392      90     A
GPO       FXI9     MX1    -E.BIL                                                FCC0392      91     A
GPO                SX0    B1          PSP = 1       */ POST STORES POSSIBLE     FCC0392      92     A
GPO                BX6    -X1*X5      I = BI[XW]                                FCC0392      93     A
GPO                SB2    X6                                                    FCC0392      94     A
GPO                SA6    A5          [XI] = I      */ RESTORE *BTT* ENTRY      FCC0392      95     A
GPO                BX7    X5                                                    FCC0392      96     A
GPO                MX4    -E.NPREDL                                             FCC0392      97     A
GPO                LX7    -E.NPREDP                                             FCC0392      98     A
GPO                BX7    -X4*X7                                                FCC0392      99     A
GPO                ZR     X7,FXI11    IF NPRED[XW] = 0                          FCC0392     100     A
GPO                SA1    B6+B2                                                 FCC0392     101     A
GPO                LX1    59-BI.HNP                                             FCC0392     102     A
GPO                PL     X1,FXI10    IF ^HN[BIT(I)]                            FCC0392     103     A
GPO                                                                             FCC0392     104     A
GPO      *         SUCCESSOR NODE IS THE HEADER NODE OF ANOTHER LOOP, SEARCH *HNT  FCC0392     105     A
GPO      *         FOR I, AND SUBSTITUTE THE *BI* OF THE *HB*.                   FCC0392     106     A
GPO                                                                             FCC0392     107     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                SA2    L.HNT                                                FCC0392      108     A
GPO                SA1    O.HNT                                                FCC0392      109     A
GPO                SA6    X2          [L.HNT] = I   */ SEARCH TERMINATOR       FCC0392      110     A
GPO                SA1    X1          HI = [O.HNT]                             FCC0392      111     A
GPO                                                                            FCC0392      112     A
GPO       FXI9A    BX2    X1          HIL = [HI]                               FCC0392      113     A
GPO                SB3    X1                                                   FCC0392      114     A
GPO                SA1    A1+1        HI = HI + 1                              FCC0392      115     A
GPO                NE     B2,B3,FXI9A IF HNI[HIL] " I                          FCC0392      116     A
GPO                                                                            FCC0392      117     A
GPO                LX2    -H.HBIP+1                                            FCC0392      118     A
GPO                SB2    X2          I = 2*HBI[HIL]  */ SUBSTITUTE *HB*       FCC0392      119     A
GPO                MX4    -E.BIL                                               FCC0392      120     A
GPO                SX6    X2                                                   FCC0392      121     A
GPO                BX5    X4*X5                                                FCC0392      122     A
GPO                BX5    X5+X6       BI[XW] = I                               FCC0392      123     A
GPO                LX2    H.HBIP-1-H.NIPP                                      FCC0392      124     A
GPO                SX2    X2                                                   FCC0392      125     A
GPO                PRINT FXI,(* HEADER,HB,N.PRED,NIP =*4Z7),(B3,B2,X7,X2)      FCC0392      126     A
GPO                IX7    X7-X2                                                FCC0392      127     A
GPO                ZR     X7,FXI11    IF PRED[XW] = NIP[HIL]                   FCC0392      128         I
          -FCC0985
GPO                ZR     X7,FXI11    IF NPRED[XW] = NIP[HIL]                  FCC0985       10     A
GPO                                                                            FCC0392      129     A
GPO       FXI10    BX7    X5                                                   FCC0392      130     A
GPO                SX0    B0          PSP = 0                                  GPO         3019     A
GPO                LX7    59-E.JPP                                             GPO         3020     A
GPO                MI     X7,FXI11    IF JP[XW]     */ REACHED BY A JUMP       GPO         3021     A
GPO                SA1    N.LJ                                                 GPO         3022         I
          -FCC0985
GPO                SX2    X1-1                                                 GPO         3023         I
          -FCC0985
GPO                NZ     X2,FXI11    IF N.LJ " 1                              GPO         3024         I
          -FCC0985
GPO                                                                            GPO         3025     A
GPO       *        SUCC REACHED BY THE FALL THROUGH AND HAS AN ACTIVE INITIAL GPO         3026     A
GPO       *        LABEL, SETUP TO ISSUE POST STORES AFTER THE LOOP BACK JUMP. GPO        3027     A
GPO                                                                            GPO         3028     A
GPO                SA1    O.RXI                                                GPO         3029         I
          -FCC0985
GPO                SA1    EPSI                                                 FCC0985       11     A
GPO        .SCR    BIT    E.SCRP-E.ALLPP                                       GPO         3030     A
GPO                SX0    .SCR        PSP = 4       */ POST STORES POSSIBLE    GPO         3031     A
GPO                SX2    A7+2                                                 GPO         3032         I
          -FCC0985
GPO                IX6    X2-X1       EPSI = RI+2 - O.RXI  */ INDEX TO BIT VECTOR GPO        3033         I
          -FCC0985
GPO                SA6    EPSI                                                 GPO         3034         I
          -FCC0985
GPO                SX6    X1+B1       EPSI = EPSI + 1                          FCC0985       12     A
GPO                SA6    A1                                                   FCC0985       13     A
GPO                                                                            GPO         3035     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO       FXI11    LX0    E.ALLPP                                          GPO        3036    A
GPO                BX7    X0+X5       ALLP[XW] = PSP & 1                   GPO        3037    A
GPO                MX1    -E.BIL                                           GPO        3038         I
          -FCC0392
GPO                SA7    A7+B1       RI = RI + 1;  [RI] = XW              GPO        3039         I
          -FCC0392
GPO                BX6    -X1*X5      I = BI[XW]                           GPO        3040         I
          -FCC0392
GPO                SA6    A5          [XI] = I      */ RESTORE TABLE ENTRY GPO        3041         I
          -FCC0392
GPO                SA7    A7+B1       RI = RI + 1;  [RI] = XW              FCC0392     131    A
GPO                SA3    B6+B2                                            FCC0392     132    A
GPO                SA2    O.BVT                                            GPO        3042    A
GPO                LX5    -E.LINKP                                         GPO        3043    A
GPO                SA3    B6+X6                                            GPO        3044         I
          -FCC0392
GPO                LX3    -BI.BVIP                                         GPO        3045    A
GPO                SB2    X2                                               GPO        3046    A
GPO                SB3    B0          J = 0                                GPO        3047    A
GPO                SB2    B2+X3       BVA = O.BVT + BVI[BIT(I)]            GPO        3048    A
GPO                                                                        GPO        3049    A
GPO       FXI12    SA1    B2          DV = BVA                             GPO        3050    A
GPO                SA2    B2+B4       UFA = BVA + VL                       GPO        3051    A
GPO                SA3    A2+B5       BXW = [UFA+2*VL]  */ BLOCK LIVE EXIT GPO        3052    A
GPO                SA4    B7+B3       RLX = LXA + J                        GPO        3053    A
GPO                BX6    -X1*X3                                           GPO        3054    A
GPO                BX7    X6+X2       LEW = ^DEF&BXW!UFW                   GPO        3055    A
GPO                SA7    A7+B1       RI = RI + 1;  [RI] = LEW             GPO        3056    A
GPO                BX6    X4+X7       RLX = LEW ! RLX                      GPO        3057    A
GPO                SA6    A4                                               GPO        3058    A
GPO                NZ     X0,FXI13    IF PSP " 0                           GPO        3059    A
GPO                SA2    A0+B3       MDW = [MDA+J]                        GPO        3060    A
GPO                BX6    -X7*X2      MDW = MDW & ^LEW                     GPO        3061    A
GPO                SA6    A2                                               GPO        3062    A
GPO                                                                        GPO        3063    A
GPO       FXI13    SB3    B3+B1       J = J + 1                            GPO        3064    A
GPO                SB2    B2+B1       BVA = BVA + 1                        GPO        3065    A
GPO                LT     B3,B4,FXI12 IF J < VL                            GPO        3066    A
GPO                                                                        GPO        3067    A
GPO                SX4    X5          XI = LINK[XW]                        GPO        3068    A
GPO                SA5    X5          XW = [XI]                            GPO        3069    A
GPO                NZ     X4,FXI10    IF XI " 0     */ NOT END OF EXIT LIST GPO       3070         I
          -FCC0392
GPO                NZ     X4,FXI9     IF XI " 0     */ NOT END OF LIST     FCC0392     133    A
GPO                                                                        GPO        3071    A
GPO       FXI15    MX7    0                                                GPO        3072    A
GPO                SA7    A7+B1       RI = RI + 1;  [RI] = 0               GPO        3073    A
GPO                                                                        GPO        3074    A
GPO                IF     DEF,/DEBUG/FXI,2                                 GPO        3075    A
GPO                DCALL  PRNTRXI,([O.RXI])                                GPO        3076    A
GPO                PRNTABV  FXI,(LEA,MDA)                                  GPO        3077    A
GPO                EQ     FXI                                              GPO        3078    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO       SHB      TITLE  SHB - SQUEEZE HOLDING BLOCK                                GPO        3079     A
GPO      **        SHB - SQUEEZE HOLDING BLOCK TO REMOVE REDUNDANT INSTRUCTIONS.     GPO        3080     A
GPO      *                                                                          GPO        3081     A
GPO      *         ENTRY  (X6) = CT = 0 FOR FIRST SQZ, 1 FOR VAR INCR SQZ            GPO        3082     A
GPO                                                                                 GPO        3083     A
GPO       SHB      ROUTINE                                                           GPO        3084     A
GPO                SX7    0           BSI = 0                                        GPO        3085     A
GPO                SA7    BSI                                                        GPO        3086     A
GPO                ZR     X6,SHB1     IF CT = 0                                      GPO        3087     A
GPO                                                                                 GPO        3088     A
GPO      *         SETUP FOR SQUEEZE OF VARIABLE INCREMENT CODE                      GPO        3089     A
GPO                                                                                 GPO        3090     A
GPO                SA7    L.TXT       L.TXT = 0                                      GPO        3091     A
GPO                CALL   RTB         GET HOLDING BLOCK POINTERS                     GPO        3092          I
         -CCGNC
GPO                CALL   RTB#        GET HOLDING BLOCK POINTERS                     CCGNC       635     A
GPO                SA5    L.HB                                                       GPO        3093     A
GPO                SA4    L.MOD                                                      GPO        3094     A
GPO                IX2    X4+X5                                                      GPO        3095     A
GPO                ALLOC  TXT,X2-4    ALLOC( TXT , L.HB + L.MOD-4 )                  GPO        3096     A
GPO                IX0    X2+X5       FM = O.TXT + L.HB                              GPO        3097     A
GPO                SX3    X2                                                         GPO        3098     A
GPO                SA2    O.SEQ                                                      GPO        3099     A
GPO                MOVE   X5,X2,X3    MOVE( L.HB , O.SEQ , O.TXT )                   GPO        3100     A
GPO                SA1    L.MOD                                                      GPO        3101     A
GPO                SA2    O.MOD                                                      GPO        3102     A
GPO                MOVE   X1-4,X2+4,X0      MOVE( L.MOD-4 , O.MOD+4 , FM )            GPO        3103     A
GPO                SX6    4                                                          GPO        3104     A
GPO                SA6    L.MOD       L.MOD = 4                                      GPO        3105     A
GPO                                                                                 GPO        3106     A
GPO      *         TERMINATE THE HB WITH AN *EOQ* AND CALL *SQZ* TO REDUCE IT        GPO        3107     A
GPO                                                                                 GPO        3108     A
GPO       SHB1     ALLOC  TXT,4                                                      GPO        3109     A
GPO                SB7    X2          TB = O.TXT; TI = OLD(L.TXT)                    GPO        3110     A
GPO                MX6    0                                                          GPO        3111     A
GPO                SB2    B0                                                         GPO        3112     A
GPO                MX7    0                                                          GPO        3113     A
GPO                RJ     SRI         SRI( EOQ  0,0 )                                GPO        3114     A
GPO                SX6    4                                                          GPO        3115     A
GPO                SX7    B6-8                                                       GPO        3116     A
GPO                ZR     X7,SHB2     IF L.TXT = 8  */ NOTHING MOVED TO HB           GPO        3117     A
GPO                SX6    1                                                          GPO        3118     A
GPO                CALL   SQZB        SQUEEZE THE BLOCK                              GPO        3119          I
         -CCGNC
GPO                CALL   SQZB#       SQUEEZE THE BLOCK                              CCGNC       636     A
GPO                SA1    L.TXT                                                      GPO        3120     A
GPO                SX6    X1-4                                                       GPO        3121     A
GPO                                                                                 GPO        3122     A
GPO       SHB2     SA2    O.TXT                                                      GPO        3123     A
GPO                SX7    X6+4                                                       GPO        3124     A
GPO                SA6    L.HB        L.HB = L.TXT - 4                               GPO        3125     A
GPO                LX7    R2.TXTLP    TXTL[TXT+1] = L.TXT                            GPO        3126     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                SA7    X2+B1                                                GPO        3127     A
GPO                CALL   WMB         WRITE HOLDING BLOCK TO *BLK*             GPO        3128          I
          -CCGNC
GPO                CALL   WMB#        WRITE HOLDING BLOCK TO *BLK*             CCGNC       637     A
GPO                                                                            GPO        3129     A
GPO                TRACE  SHB,RLIST                                            GPO        3130     A
GPO                EQ     SHB                                                  GPO        3131     A
GPO       MII      TITLE  MII - MARK INVARIANT INSTRUCTIONS                    GPO        3132     A
GPO      **        MII - MARK INVARIANT INSTRUCTIONS, COLLECT INFORMATION FOR  GPO        3133     A
GPO      *                STRENGTH REDUCTION.                                 GPO        3134     A
GPO      *                                                                    GPO        3135     A
GPO      *         EXIT   [TBA+3] = 12/P(NI),12/0,18/STC,18/ICC               GPO        3136     A
GPO      *                                                                    GPO        3137     A
GPO      *         REGISTER ASSIGNMENTS FOR MAIN LOOP OF *MII*                GPO        3138     A
GPO      *                                                                    GPO        3139     A
GPO      *                (A0) = FAC = FUNCTION ARGUMENT CHAIN                GPO        3140     A
GPO      *                (A5) = R1 , OF CURRENT INSTRUCTION                  GPO        3141     A
GPO      *                (A7) = LII , ADDRESS OF LAST INVARIANT INSTRUCTION  GPO        3142     A
GPO      *                (B3) = 3                                            GPO        3143     A
GPO       SC       MICRO  1,,/B4/     STORE CHAIN ( POSSIBLE RECURSIVE DEFS )  GPO        3144     A
GPO       NI       MICRO  1,,/B5/     N.INCREMENT STORES ( *RD* )              GPO        3145     A
GPO       UB       MICRO  1,,/B6/     O.UDT                                    GPO        3146     A
GPO                                                                            GPO        3147     A
GPO                QUAL   MII                                                  GPO        3148     A
GPO                                                                            GPO        3149     A
GPO       MII      ROUTINE                                                     GPO        3150     A
GPO                BX6    X4                                                   GPO        3151     A
GPO                LX6    59-IM.APP                                            GPO        3152     A
GPO                SA6    AN          AN = AP[BSW]  */ SET ARTICULATION NODE INDI  GPO   3153     A
GPO                                                                            GPO        3154     A
GPO                LX4    -IM.BIP                                              GPO        3155     A
GPO                FBVA   X4          BVA = O.BVT + BVI[BIT(BN)]               GPO        3156     A
GPO                SX7    X7+B2                                                GPO        3157     A
GPO                SA7    UBD         UBD = BVA + VL                           GPO        3158     A
GPO                SB3    3                                                    GPO        3159     A
GPO                SA4    O.UDT                                                GPO        3160     A
GPO                S"TB"  X5+B3       TB = TBA + 3  */ LINK WORD OF BASE       GPO        3161     A
GPO                SA0    B0          FAC = 0       */ FUNCTION ARG CHAIN      GPO        3162     A
GPO                S"UB"  X4          UB = [O.UDT]                             GPO        3163     A
GPO                SA5    X5          R1 = TBA                                 GPO        3164     A
GPO                MX7    IM.APLNL                                             GPO        3165     A
GPO                BX3    X7*X6                                                GPO        3166     A
GPO                LX3    IM.APLNL+R1.H2P                                      GPO        3167     A
GPO                BX7    X3+X5       H2[R1] = APLN[BSW]  */ SAVE APLN FOR *GRA*  GPO       3168     A
GPO                SA7    A5                                                   GPO        3169     A
GPO                MX6    0                                                    GPO        3170     A
GPO                S"SC"  B0          SC = 0        */ STORE CHAIN             GPO        3171     A
GPO                S"NI"  B0          NI = 0        */ N. RECURSIVE DEFINITIONS GPO        3172     A
GPO                SA1    "TB"+B1                                              GPO        3173     A
GPO                MX7    1                                                    GPO        3174     A
GPO                SA6    IJP         IJP = 0       */ INVARIANT JUMP FLAG     GPO        3175     A
GPO                SA6    A6+B1       UIC = 0       */ UNSAFE INV CODE FLAG    GPO        3176     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO              LX7    1+I.INVP                                          GPO        3177    A
GPO              SA7    "TB"        LII = TB;  [LII] = *INV */ LAST INV INST  GPO   3178    A
GPO                                                                       GPO        3179    A
GPO              UX6    B2,X1                                             GPO        3180    A
GPO              SX7    B2-OC.LAB                                         GPO        3181    A
GPO              NZ     X7,MII2     IF OC[R1+4] " OC.LAB                  GPO        3182    A
GPO                                                                       GPO        3183    A
GPO              SA5    A1          R1 = R1 + 4   */ SKIP INITIAL LABEL   GPO        3184    A
GPO              EQ     MII2                                              GPO        3185    A
GPO       MII    SPACE  2,14                                              GPO        3186    A
GPO      *       OPERANDS *INV*, MARK INSTRUCTION AS *INV*                GPO        3187    A
GPO                                                                       GPO        3188    A
GPO       MII1   SA2    A5+2        DI = R1 + 2                           GPO        3189    A
GPO              SA1    AN                                                GPO        3190    A
GPO              LX2    59-D.USIP                                         GPO        3191    A
GPO              BX3    -X2+X1                                            GPO        3192    A
GPO              MI     X3,MII1A    IF AN ! ^USI[DI]  */ SAFE TO MOVE     GPO        3193    A
GPO              SX6    1                                                 GPO        3194    A
GPO              SA6    UIC         UIC = 1       */ INDICATE UNSAFE INV CODE  GPO  3195    A
GPO              EQ     MII2                                              GPO        3196    A
GPO                                                                       GPO        3197    A
GPO              PROCESS LDC                                              CCG1         87          I
         -CCG16
GPO              PROCESS (LDC,LDV)                                        CCG16       203    A
GPO       MII1A  SX6    A7-"TB"     IIC[R1+3] = LII - TB                  GPO        3198    A
GPO              MX4    1                                                 GPO        3199    A
GPO              LX6    I.IICP                                            GPO        3200    A
GPO              LX4    1+I.INVP    INV[R1+3] = 1                         GPO        3201    A
GPO              BX7    X4+X6                                             GPO        3202    A
GPO              SA7    A5+B3       LII = R1 + 3                          GPO        3203    A
GPO                                                                       GPO        3204    A
GPO      *       GET NEXT INSTRUCTION AND JUMP TO PROCESSOR               GPO        3205    A
GPO                                                                       GPO        3206    A
GPO              PROCESS (DAR,DEF,NOP)                                    GPO        3207    A
GPO       MII2   SA5    A5+4        R1 = R1 + 4                           GPO        3208    A
GPO              UX6    B2,X5                                             GPO        3209    A
GPO              JP     MII.JT+B2   JUMP( MII.JT( OC[R1] )                GPO        3210    A
GPO       TYII   SPACE  2                                                 GPO        3211    A
GPO       TYII   EQU    MII1A                   */ TYPE II ARE *INV*     GPO        3212    A
GPO                                                                       GPO        3213    A
GPO       TYIV   EQU    MII2                    */ TYPE IV ARE NOT INV   GPO        3214    A
GPO       LD     SPACE  3,14                                              GPO        3215    A
GPO      *       LD - CHECK *UDI* TO SEE IF *INV* IN REGION               GPO        3216    A
GPO                                                                       GPO        3217    A
GPO              PROCESS (LD,ILD)                                         GPO        3218         I
         -FCC0885
GPO              PROCESS ILD                                              FCC0885       6    A
GPO              SB2    OC.LD                                             FCC0885       7         I
         -FCC0885XA
GPO              PX6    B2,X5       OC[R1] = OC.LD  */ INHIBIT ST/ILD SQZ FCC0885       8         I
         -FCC0885XA
GPO              SB2    OC.LD       OC = OC.LD                            FCC0885XA     5    A

         0        1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO       .^PLI    IFNE   HC.ID,3     ^ PL/I                                          FCC0885XA      6     A
GPO                SA4    IT.                                                         FCC0885XA      7     A
GPO                SA3    A5+B1                                                       FCC0885XA      8     A
GPO                BX6    X3-X4                                                       FCC0885XA      9     A
GPO                SX7    X6                                                          FCC0885XA     10     A
GPO                NZ     X7,ILD1     IF IH[R2] " IT.                                 FCC0885XA     11     A
GPO                SB2    OC.TLD                                                      FCC0885XA     12     A
GPO                                                                                   FCC0885XA     13     A
GPO       .^PLI    ENDIF              ^ PL/I                                          FCC0885XA     14     A
GPO       ILD1     PX6    B2,X5       OC[R1] = OC   */ INHIBIT ST/ILD SQZ             FCC0885XA     15     A
GPO                SA6    A5                                                          FCC0885        9     A
GPO                                                                                   FCC0885       10     A
GPO                PROCESS LD                                                         FCC0885       11           I
          -FCCA506X
GPO                PROCESS (LD,TLD)                                                   FCCA506X       4     A
GPO                SA4    A5+B1       R2 = R1 + 1                                     GPO         3219     A
GPO                LX5    -R1.INP                                                     GPO         3220     A
GPO                SA3    "UB"+X5     UI = UB + IN[R1]                                GPO         3221     A
GPO                LX4    -IH.RFP                                                     GPO         3222     A
GPO                BX7    "RN"X4                                                      GPO         3223     A
GPO                SA2    "TB"+X7     LF = TB + RF[R2]                                GPO         3224           I
          -FCC1504X
GPO                NZ     X7,LD1      IF RF[R2] " 0                                   FCC1504X      10     A
GPO                SA2    A4+B1       DI = R2 + 1                                     FCC1504X      11     A
GPO                SX6    B1                                                          FCC1504X      12     A
GPO                LX2    -D.RFP                                                      FCC1504X      13     A
GPO                BX6    X6*X2                                                       FCC1504X      14     A
GPO                LX6    UD.RFP                                                      FCC1504X      15     A
GPO                BX6    X6+X3       RF[UI] = RF[UI] ! RF[DI]                        FCC1504X      16     A
GPO                SA6    A3                                                          FCC1504X      17     A
GPO       LD1      SA2    "TB"+X7     LF = TB + RF[R2]                                FCC1504X      18     A
GPO                LX3    59-UD.DEFRP                                                 GPO         3225     A
GPO                LX2    59-I.INVP                                                   GPO         3226     A
GPO                BX6    -X3*X2                                                      GPO         3227     A
GPO                PL     X6,MII2     IF ^( ^DEFR[UI] & INV[LF] )                     GPO         3228     A
GPO                LX3    UD.DEFRP-UD.ISTP                                  000200    FCC1040A       4     A
GPO                MI     X3,MII2     IF IST[UI]    */ INTERFERING STORE        000210    FCC1040A       5     A
GPO                SA3    AN                                                          FCC1584X       5     A
GPO                MI     X3,MII1A    IF AN < 0     */ SAFE TO MOVE                   FCC1584X       6     A
GPO                ZR     X7,MII1A    IF RF[R2] = 0 */ NO SUBSCRIPT                   FCC1584X       7     A
GPO                SA3    =XHO$UO
>>>>      CCGA029 11                                                                  CCGA029        5     A
GPO                ZR     X3,MII2     IF OPT LT 3   */ DONT MOVE A(I)
>>>>      CCGA029 12                                                                  CCGA029        6     A
GPO                SA3    A2-B3       R1F = LF - 3                                    FCC1584X       8     A
GPO                UX7    B2,X3                                                       FCC1584X       9     A
GPO                SX6    B2-OC.LD                                                    FCC1584X      10     A
GPO                NZ     X6,MII1A    IF OC[R1F] " OC.LD                              FCC1584X      11     A
GPO                SA2    A3+B1                                                       FCC1584X      12     A
GPO                LX2    -IH.RFP                                                     FCC1584X      13     A
GPO                BX7    "RN"X2                                                      FCC1584X      14     A
GPO                NZ     X7,MII2     IF RF[R2F] " 0  */ DONT MOVE A(B(I))            FCC1584X      15     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                EQ     MII1                                                GPO          3229     A
GPO        TYI     SPACE  3,14                                                GPO          3230     A
GPO       *        TYPE I , INV(I) = INV(J) & INV(K)                          GPO          3231     A
GPO                                                                           GPO          3232     A
GPO        TYI     LX5    -R1.RJP                                             GPO          3233     A
GPO                BX6    "RN"X5                                              GPO          3234     A
GPO                LX5    R1.RJP-R1.RKP                                       GPO          3235     A
GPO                SA4    "TB"+X6     LJ = TB + RJ[R1]                        GPO          3236     A
GPO                BX7    "RN"X5                                              GPO          3237     A
GPO                SA3    "TB"+X7     LK = TB + RK[R1]                        GPO          3238     A
GPO                BX6    X3*X4                                               GPO          3239     A
GPO                LX6    59-I.INVP                                           GPO          3240     A
GPO                MI     X6,MII1     IF INV[LJ] & INV[LK]                    GPO          3241     A
GPO                EQ     MII2                                                GPO          3242     A
GPO                                                                           GPO          3243     A
GPO       *        TYPE 1 - 2 RESULTS                                         GPO          3244     A
GPO                                                                           GPO          3245     A
GPO                PROCESS (NR,UP,RNZ)                                        GPO          3246     A
GPO                LX5    -R1.RKP                                             GPO          3247     A
GPO                BX6    "RN"X5                                              GPO          3248     A
GPO                SA4    "TB"+X6     LK = TB + RK[R1]                        GPO          3249     A
GPO                LX4    59-I.INVP                                           GPO          3250     A
GPO                MI     X4,MII1     IF INV[LK]                              GPO          3251     A
GPO                EQ     MII2                                                GPO          3252     A
GPO                                                                           GPO          3253     A
GPO       *        CONSTANT SHIFTS                                            GPO          3254     A
GPO                                                                           GPO          3255     A
GPO                PROCESS (KLS,KRS)                                          GPO          3256     A
GPO                SA4    A5+B1       R2 = R1 + 1                             GPO          3257     A
GPO                LX4    -IH.RFP                                             GPO          3258     A
GPO                SA3    "TB"+X4     LF = TB + RF[R2]                        GPO          3259     A
GPO                LX3    59-I.INVP                                           GPO          3260     A
GPO                MI     X3,MII1     IF INV[LF]                              GPO          3261     A
GPO                EQ     MII2                                                GPO          3262     A
GPO        IM      SPACE  2,14                                                FCC0238AX       6     A
GPO       **       IM - MAY BE UNSAFE IF CONDITIONALLY EXECUTED               FCC0238AX       7     A
GPO                                                                           FCC0238AX       8     A
GPO                PROCESS IM                                                 FCC0238AX       9     A
GPO                LX5    -R1.RJP                                             FCC0238AX      10     A
GPO                BX6    "RN"X5                                              FCC0238AX      11     A
GPO                LX5    R1.RJP-R1.RKP                                       FCC0238AX      12     A
GPO                SA4    "TB"+X6     LJ = TB + RJ[R1]                        FCC0238AX      13     A
GPO                BX7    "RN"X5                                              FCC0238AX      14     A
GPO                SA3    "TB"+X7     LK = TB + RK[R1]                        FCC0238AX      15     A
GPO                BX6    X3*X4                                               FCC0238AX      16     A
GPO                LX6    59-I.INVP                                           FCC0238AX      17     A
GPO                PL     X6,MII2     IF ^(INV[LJ] & INV[LK])                 FCC0238AX      18     A
GPO                SA2    AN                                                  FCC0238AX      19     A
GPO                MI     X2,MII1A    IF AN < 0     */ UNCONDITIONALLY EXECUTED FCC0238AX      20     A
GPO                                                                           FCC0238AX      21     A
GPO       *        INV AND CONDITIONALLY EXECUTED, MARK AS *USI* SO *SQZ* WILL FCC0238AX      22     A
GPO       *        ADD A UNPACK OF VARIABLE OPERANDS PRIOR TO THE MULTIPLY.   FCC0238AX      23     A

          0        1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO       *         NOTE *LD* IS UNSAFE IF IT IS A PROGRAMMER VAR, NOT A VD.      FCC0238AX     24     A
GPO                                                                               FCC0238AX     25     A
GPO                 SA2    A4-B1       DJ = LJ - 1                                FCC0238AX     26     A
GPO                 CALL   CMO                                                    FCC0238AX     27     A
GPO                 SA2    A3-B1       DK = LK - 1                                FCC0238AX     28     A
GPO                 ZR     X6,IM1      IF CMO(DJ) = 0  */ J OP IS SAFE            FCC0238AX     29     A
GPO                 CALL   CMO                                                    FCC0238AX     30     A
GPO                 ZR     X6,IM2      IF CMO(DK) = 0  */ K OP IS SAFE            FCC0238AX     31     A
GPO                 EQ     MII2                                                   FCC0238AX     32     A
GPO                                                                               FCC0238AX     33     A
GPO        IM1      CALL   CMO                                                    FCC0238AX     34     A
GPO                 ZR     X6,MII1A    IF CMO(DK) = 0  */ K OP IS SAFE            FCC0238AX     35     A
GPO                                                                               FCC0238AX     36     A
GPO        IM2      SA4    A5+2        DI = R1 + 2                                FCC0238AX     37     A
GPO                 SX3    B1                                                     FCC0238AX     38     A
GPO                 LX3    D.USIP                                                 FCC0238AX     39     A
GPO                 BX6    X3+X4       USI[DI] = 1   */ MARK AS UNSAFE            FCC0238AX     40     A
GPO                 SA6    A4                                                     FCC0238AX     41     A
GPO                 EQ     MII1A                                                  FCC0238AX     42     A
GPO        ADDS     SPACE  3,14                                                   GPO         3263     A
GPO       *         STT , IA, IS - CHECK FOR =(I+INV)                             GPO         3264     A
GPO                                                                               GPO         3265     A
GPO                 PROCESS STT                                                   GPO         3266     A
GPO                 SA3    A5+B1       R2 = R1 + 1                                GPO         3267     A
GPO                 LX3    -IH.IHP                                                GPO         3268     A
GPO                 SB2    X3                                                     GPO         3269     A
GPO                 LX3    IH.IHP-IH.RFP                                          GPO         3270     A
GPO                 SA4    "TB"+X3     LJ = TB + RF[R2]                           GPO         3271     A
GPO                 LX4    59-I.INVP                                              GPO         3272     A
GPO                 MI     X4,MII1     IF INV[LJ]                                 GPO         3273     A
GPO                                                                               GPO         3274     A
GPO                 SA1    A4-3        R1P = LJ - 3                               GPO         3275     A
GPO                 ZR     B2,IA2      IF IH[R2] = 0                              GPO         3276     A
GPO                                                                               GPO         3277     A
GPO                 EQ     MII2                                                   GPO         3278     A
GPO        ADDS     EJECT                                                         GPO         3279     A
GPO                 PROCESS (IA,IS)                                               GPO         3280     A
GPO                 LX5    -R1.RJP                                                GPO         3281     A
GPO                 BX6    "RN"X5                                                 GPO         3282     A
GPO                 LX5    R1.RJP-R1.RKP                                          GPO         3283     A
GPO                 SA4    "TB"+X6     LJ = TB + RJ[R1]                           GPO         3284     A
GPO                 BX7    "RN"X5                                                 GPO         3285     A
GPO                 SA3    "TB"+X7     LK = TB + RK[R1]                           GPO         3286     A
GPO                 BX6    X3*X4                                                  GPO         3287     A
GPO                 LX6    59-I.INVP                                              GPO         3288     A
GPO                 BX7    X3+X4                                                  GPO         3289     A
GPO                 MI     X6,MII1     IF INV[LJ] & INV[LK]                       GPO         3290          I
          -FCC0716
GPO                 MI     X6,MII1A    IF INV[LJ] & INV[LK]                       FCC0716        9     A
GPO                 LX7    59-I.INVP                                              GPO         3291     A
GPO                 PL     X7,MII2     IF ^( INV[LJ] ! INV[LK] )                  GPO         3292     A
GPO                                                                               GPO         3293     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO              SX7    A4-B3       R1P = LJ - 3                                    GPO        3294    A
GPO              SA2    A3-B1       DP = LK - 1                                     GPO        3295    A
GPO              LX4    59-I.INVP                                                   GPO        3296    A
GPO              PL     X4,IA1      IF ^INV[LJ]                                     GPO        3297    A
GPO              SB2    B2-OC.IS                                                    GPO        3298    A
GPO              ZR     B2,MII2     IF OC[R1] = OC.IS  */ INV - I CASE              GPO        3299    A
GPO              SX7    A3-B3       R1P = LK - 3                                    GPO        3300    A
GPO              SA2    A4-B1       DP = LJ - 1                                     GPO        3301    A
GPO                                                                                GPO        3302    A
GPO       IA1    LX2    59-D.ZPP                                                    GPO        3303    A
GPO              SA1    X7                                                          GPO        3304    A
GPO              PL     X2,MII2     IF ^ZP[DP]    */ INV OP IS ^ZP                  GPO        3305         I
          -CCG36
GPO              PL     X2,MII2     IF ^ZP[DP]    */ INV OP IS ^ZP          000190  CCG36         5    A
GPO              SA2    A2-2        R1P = DP - 2                            000200  CCG36         6    A
GPO              UX6    B2,X2                                               000210  CCG36         7    A
GPO              SX6    B2-OC.FMA                                           000220  CCG36         8    A
GPO              NZ     X6,IA1A     IF OC[R1P] " OC.FMA                     000230  CCG36         9    A
GPO              LX2    -R1.INP                                             000240  CCG36        10    A
GPO              SX6    X2-37                                               000250  CCG36        11    A
GPO              MI     X6,MII2     IF IN[R1P] < 37  */ MASK VALUE TOO BIG  000260  CCG36        12    A
GPO              EQ     IA2                                                 000270  CCG36        13    A
GPO                                                                         000280  CCG36        14    A
GPO       IA1A   SX6    B2-OC.LDC                                           000290  CCG36        15    A
GPO              NZ     X6,IA2      IF OC[R1P] " OC.LDC                     000300  CCG36        16    A
GPO              SA2    A2+B1       R2P = R1P + 1                           000310  CCG36        17    A
GPO              SA3    =XO$CVT                                             000320  CCG36        18    A
GPO              LX2    -IH.CAP                                             000330  CCG36        19    A
GPO              SB2    X3                                                  000340  CCG36        20    A
GPO              SA2    B2+X2       C = CVT(CA[R2+])                        000350  CCG36        21    A
GPO              AX2    23                                                  000360  CCG36        22    A
GPO              NZ     X2,MII2     IF ABS(C) @ 2**22  */ INCR TOO BIG      000370  CCG36        23    A
GPO                                                                                GPO        3306    A
GPO       *      ONE OPERATION IS INV, SEE IF THE OTHER IS A *LD* OF A *RD* VAR     GPO        3307    A
GPO                                                                                GPO        3308    A
GPO       IA2    UX7    B2,X1                                                       GPO        3309    A
GPO              SX4    B2-OC.LD                                                    GPO        3310    A
GPO              LX1    -R1.INP                                                     GPO        3311    A
GPO              NZ     X4,MII2     IF OC[R1P] " OC.LD                              GPO        3312    A
GPO              SA2    "UB"+X1                                                     GPO        3313    A
GPO              LX2    59-UD.^RDP                                                  GPO        3314    A
GPO              MI     X2,MII2     IF ^RD[ UDI(IN[R1P]) ]                          GPO        3315    A
GPO                                                                                GPO        3316    A
GPO              SX5    B1                                                          GPO        3317    A
GPO              SX1    X1                                                          GPO        3318    A
GPO              LX5    I.IAP                                                       GPO        3319    A
GPO              LX1    I.LDIP      LDI[R1+3] = IN[R1]                              GPO        3320    A
GPO              BX6    X5+X1                                                       GPO        3321    A
GPO              SA6    A5+3        IA[R1+3] = 1  */ INDICATE  =(I+CON)             GPO        3322    A
GPO              EQ     MII2                                                        GPO        3323    A
GPO       UJP    SPACE  2,14                                                        GPO        3324    A
GPO       *      JUMP PROCESSING, CHECK FOR LABEL SUBSTITUTION, BUMP N.LJ           GPO        3325    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                                                                                GPO        3326    A
GPO                PROCESS UJP                                                     GPO        3327    A
GPO                SA3     LIH                                                     GPO        3328    A
GPO                BX4     X5-X3                                                   GPO        3329    A
GPO                SB2     X4                                                      GPO        3330    A
GPO                NZ      B2,JPX2     IF IH[R1] " LIH                             GPO        3331    A
GPO                                                                                GPO        3332    A
GPO                SA2     N.LJ                                                    GPO        3333    A
GPO                SA3     LGL                                                     GPO        3334    A
GPO                MX7     1                                                       GPO        3335    A
GPO                BX6     X7+X2       N.LJ = N.LJ ! 1S59  */ INHIBIT PREFETCHING  GPO        3336    A
GPO                SA6     A2                                                      GPO        3337    A
GPO                ZR      X3,JPX2     IF LGL = 0    */ NO LABEL SUBSTITUTION      GPO        3338    A
GPO                AX3     18                                                      GPO        3339    A
GPO                IX6     X3+X5       IH[R1] = SHIFT( LGL , -18 )                 GPO        3340    A
GPO                SA6     A5                                                      GPO        3341    A
GPO                EQ      JPX2                                                    GPO        3342    A
GPO                                                                                GPO        3343         I
          -FCC0909
GPO        JPX     SPACE   2                                                       FCC0909       8    A
GPO                PROCESS JPX                                                     GPO        3344    A
GPO                SA4     A5+B1       R2 = R1 + 1                                 GPO        3345    A
GPO                SA3     LIH                                                     GPO        3346    A
GPO                BX7     X3-X4                                                   GPO        3347    A
GPO                SB2     X7                                                      GPO        3348    A
GPO                NZ      B2,JPX1     IF IH[R2] " LIH                             GPO        3349    A
GPO                                                                                GPO        3350    A
GPO                SA2     N.LJ                                                    GPO        3351    A
GPO                SA3     LGL                                                     GPO        3352    A
GPO                SX6     X2+B1       N.LJ = N.LJ + 1                             GPO        3353         I
          -FCCA462
GPO                SX7     B1                                          002480      FCCA462       9    A
GPO                IX6     X7+X2       N.LJ = N.LJ + 1                 002490      FCCA462      10    A
GPO                SA6     A2                                                      GPO        3354    A
GPO                ZR      X3,JPX1     IF LGL = 0    */ NO LABEL SUBSTITUTION      GPO        3355    A
GPO                AX3     18                                                      GPO        3356    A
GPO                IX6     X4+X3       IH[R2] = SHIFT(LGL,-18)                     GPO        3357    A
GPO                SA6     A4+                                                     GPO        3358    A
GPO                                                                                GPO        3359    A
GPO        JPX1    BX6     "RN"X5                                                  GPO        3360    A
GPO                SA4     "TB"+X6     LI = TB + RI[R1]                            GPO        3361    A
GPO                LX4     59-I.INVP                                               GPO        3362    A
GPO                PL      X4,JPX2     IF ^INV[LI]                                 GPO        3363    A
GPO                                                                                GPO        3364    A
GPO                SA6     IJP         IJP = RI[R1]  */ SAVE OPERAND ADDR OF JP    GPO        3365    A
GPO                                                                                GPO        3366    A
GPO       *        LAB, JUMP - CLEAR *AN*, IN CASE OPT=1 ( NO CONTROL FLOW INFO )  GPO        3367    A
GPO                                                                                GPO        3368    A
GPO                PROCESS (LAB,JPBB)                                              GPO        3369    A
GPO        JPX2    SX6     0                                                       GPO        3370         I
          -CCG34
GPO                SA6     AN          AN = 0                                      GPO        3371         I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

          -CCG34
GPO        JPX2     SA3    AN                                          000780  CCG34          5     A
GPO                 MX4    1                                           000790  CCG34          6     A
GPO                 BX6    -X4*X3      AP[AN] = 0                      000800  CCG34          7     A
GPO                 SA6    A3                                          000810  CCG34          8     A
GPO                 EQ     MII2                                                GPO         3372     A
GPO        JIN      SPACE  2                                                   GPO         3373          I
          -FCC0909
GPO        JIN      SPACE  2,14                                                FCC0909        9     A
GPO       *         JIN - INDEXED JUMP ( COMPUTED OR ASSIGNED GOTO )           FCC0909       10     A
GPO                                                                            FCC0909       11     A
GPO                 PROCESS JIN                                                GPO         3374     A
GPO                 SA2    N.LJ                                                GPO         3375     A
GPO                 MX7    1                                                   GPO         3376     A
GPO                 BX6    X7+X2       N.LJ = 1S59 ! N.LJ  */ INHIBIT PREFETCHING  GPO     3377     A
GPO                 SA6    A2                                                  GPO         3378     A
GPO                 SA3    AN                                                  FCC0909       12     A
GPO                 SA2    LGL                                                 FCC0909       13     A
GPO                 LX3    IM.APP-IM.LNP                                       FCC0909       14     A
GPO                 SA4    A5+B1       R2 = R1 + 1                             FCC0909       15     A
GPO                 PL     X3,JPX2     IF ^LN[AN]                              FCC0909       16     A
GPO                 ZR     X2,JPX2     IF LGL = 0    */ NO LABEL CHANGE        FCC0909       17     A
GPO                 SB2    X4                                                  FCC0909       18     A
GPO                 NZ     B2,JPX2     IF IH[R2] " 0 */ ^ AN ASSIGNED GOTO     FCC0909       19     A
GPO                                                                            FCC0909       20     A
GPO       *         SET FLAGS ( N.LJ , B59 OF R2 WORD ) TO INDICATE TO *GRA*   FCC0909       21     A
GPO       *         THAT SPECIAL CODE MAY BE NECESSARY FOR THIS *JIN* .        FCC0909       22     A
GPO                                                                            FCC0909       23     A
GPO                 LX7    -1                                                  FCC0909       24     A
GPO                 BX6    X7+X6       N.LJ = N.LJ + 1S58                      FCC0909       25     A
GPO                 SA6    A6                                                  FCC0909       26     A
GPO                 LX7    1                                                   FCC0909       27     A
GPO                 BX6    X7+X4       [R2] = [R2] + 1S59                      FCC0909       28     A
GPO                 SA6    A4                                                  FCC0909       29     A
GPO                 EQ     JPX2                                                GPO         3379     A
GPO        RJXJ     SPACE  3                                                   CCG2           1     A
GPO       *         RJXJ - CONDITIONAL RETURN JUMP                             CCG2           2     A
GPO                                                                            CCG2           3     A
GPO                 PROCESS RJXJ                                               CCG2           4     A
GPO                 SA3    A5+B1       R2 = R1 + 1                             CCG2           5     A
GPO                 SA4    "TB"+X5     LI = TB + RI[R1]                        CCG2           6     A
GPO                 LX3    -IH.RFP                                             CCG2           7     A
GPO                 SA3    "TB"+X3     LF = TB + RF[R2]                        CCG2           8     A
GPO                 BX6    X3*X4                                               CCG2           9     A
GPO                 LX6    59-I.INVP                                           CCG2          10     A
GPO                 PL     X6,MII2     IF ^(INV[LI] & INV[LF])                 CCG2          11     A
GPO                 EQ     MII1A                                               CCG2          12     A
GPO        TST      SPACE  3,14                                                FCC1254       21     A
GPO       *         TST -  DEFINITION POINT OF MOVED INSTRUCTION               FCC1254       22     A
GPO                                                                            FCC1254       23     A
GPO                 PROCESS TST                                                FCC1254       24     A
GPO                 BX6    "RN"X5                                              FCC1254       25     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                SA4    "TB"+X6      LI = TB + RI[R1]                          FCC1254      26     A
GPO                SA3    O.TET                                                  FCC1254      27     A
GPO                LX4    59-I.INVP                                              FCC1254      28     A
GPO                PL     X4,MII2      IF ^INV[LI]                               FCC1254      29     A
GPO                                                                              FCC1254      30     A
GPO                SA1    A5+B1        R2 = R1 + 1                               FCC1254      31     A
GPO                SB2    X3                                                     FCC1254      32     A
GPO                LX1    -IH.CAP                                                FCC1254      33     A
GPO                SA2    B2+X1                                                  FCC1254      34     A
GPO                LX2    59-T.INVP                                              FCC1254      35     A
GPO                PL     X2,MII2      IF ^INV[TET(CA[R2])]                      FCC1254      36     A
GPO                EQ     MII1A                                                  FCC1254      37     A
GPO         RS     EJECT                                                         GPO        3380     A
GPO       *        REGISTER STORE PROCESSING                                     GPO        3381     A
GPO                                                                              GPO        3382     A
GPO                PROCESS RS                                                    GPO        3383     A
GPO                SA4    A5-B1        LI = R1 - 1                               GPO        3384     A
GPO                MX7    -SO.LKL                                                GPO        3385     A
GPO                LX5    -R1.SOP-SO.LKP                                         GPO        3386     A
GPO                BX6    -X7*X5       LKT = SOLK[R1]                            GPO        3387     A
GPO                LX4    59-I.INVP                                              GPO        3388     A
GPO                ZR     X6,RS1       IF LKT = 0    */ UJP LOCK                 GPO        3389     A
GPO                SB2    X6-2                                                   GPO        3390     A
GPO                LT     B2,MII2      IF LKT = 1    */ TEMP LOCK                GPO        3391     A
GPO                PL     X4,MII2      IF ^INV[LI]                               GPO        3392     A
GPO                                                                              GPO        3393     A
GPO                GT     B2,MII1      IF LKT = 3    */ *RJ RS*                  GPO        3394     A
GPO                                                                              GPO        3395     A
GPO       *        FULL LOCK *RS* AND OPERAND INVARIANT IN CURRENT LOOP          GPO        3396     A
GPO                                                                              GPO        3397     A
GPO                SX7    B1                                                     GPO        3398     A
GPO                LX5    R1.SOP+SO.LKP                                          GPO        3399     A
GPO                LX7    R1.SOP+SO.INVCP                                        GPO        3400     A
GPO                BX6    X7+X5        INVC[R1] = 1  */ INDICATE INV IN THIS LOOP GPO        3401     A
GPO                SA6    A5                                                     GPO        3402     A
GPO                MX7    -SO.RTL                                                GPO        3403     A
GPO                LX5    -R1.SOP-SO.RTP                                         GPO        3404     A
GPO                BX4    -X7*X5                                                 GPO        3405     A
GPO                ZR     X4,RS0       IF SORT[R1] = 0  */ RS TO A B-REG         GPO        3406     A
GPO                SA3    A5-B3        R2P = R1 - 3                              GPO        3407     A
GPO                LX3    59-IH.LDP                                              GPO        3408     A
GPO                PL     X3,MII2      IF ^LD[R2P]   */ MOVE IF NOT A PREFETCH   GPO        3409     A
GPO                                                                              GPO        3410     A
GPO       *        REMOVE PRED OF *RS* FROM INVARIANT CHAIN                      GPO        3411     A
GPO                                                                              GPO        3412     A
GPO         RS0    SA4    A5-B1        LIP = R1 - 1                              GPO        3413     A
GPO                LX4    -I.IICP      I = IIC[LIP]                              GPO        3414     A
GPO                SA3    "TB"+X4      LJP = TB + I                              GPO        3415     A
GPO                BX7    X3                                                     GPO        3416     A
GPO                SA7    A3           LII = LJP                                 GPO        3417     A
GPO                MX6    0                                                      GPO        3418     A
GPO                SA6    A4           [LIP] = 0                                 GPO        3419     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                EQ     MII2                                                    GPO          3420     A
GPO                                                                               GPO          3421     A
GPO       *        *UJP* RS , PRECEEDING A FUNCTION CALL                           GPO          3422     A
GPO                                                                               GPO          3423     A
GPO        RS1     MI     X4,RS2      IF INV[LI]                                   GPO          3424     A
GPO                SA0    -1          FAC = 0       */ INDICATE ^INV OPERAND       GPO          3425     A
GPO                EQ     MII2                                                    GPO          3426     A
GPO                                                                               GPO          3427     A
GPO        RS2     SX6    A0                                                      GPO          3428     A
GPO                MI     X6,MII2     IF FAC < 0    */ PREVIOUS ARG NOT INV         GPO          3429     A
GPO                LX6    I.IICP                                                  GPO          3430     A
GPO                SA6    A5+B3       IIC[R1+3] = FAC  */ CHAIN THIS TO LAST        GPO          3431     A
GPO                SA0    A6-"TB"     FAC = R1+3 - TB                             GPO          3432     A
GPO                EQ     MII2                                                    GPO          3433     A
GPO        RJ3     SPACE  3                                                       GPO          3434          I
          -FCC1254
GPO        RJ3     SPACE  2,14                                                    FCC1254        38     A
GPO       *        RJ3 - CHECK FOR *BEF* AND ARGS INVARIANT                        GPO          3435     A
GPO                                                                               GPO          3436     A
GPO                PROCESS RJ3                                                    GPO          3437          I
          -CCG1
GPO                PROCESS (RJ3,RJ6)                                              CCG1           88     A
GPO                SA1    AN                                                      GPO          3438     A
GPO                SB2    A0          AC = FAC                                    GPO          3439     A
GPO                SA0    B0          FAC = 0                                     GPO          3440     A
GPO                PL     X1,MII2     IF ^AN        */ ON A SIDE BRANCH           GPO          3441     A
GPO                LE     B2,MII2     IF AC @ 0     */ ARGS NOT INV                GPO          3442     A
GPO                SA4    A5+B1       R2 = R1 + 1                                 GPO          3443     A
GPO                MX7    -FI.FTL                                                 GPO          3444     A
GPO                LX4    -FI.FTP                                                 GPO          3445     A
GPO                BX6    -X7*X4                                                  GPO          3446     A
GPO                SX3    X6-2                                                    GPO          3447     A
GPO                NZ     X3,MII2     IF FT[R2] " 2 */ NOT A *BEF*                GPO          3448     A
GPO        RJ3     SPACE  1,10                                                    GPO          3449     A
GPO       *        MARK *RS"S* AS INV SO WE KNOW TO MOVE THEM OUT                  GPO          3450     A
GPO                                                                               GPO          3451     A
GPO        RJ31    SA3    "TB"+B2     LIP = TB + AC                               GPO          3452     A
GPO                MX4    1                                                       GPO          3453     A
GPO                LX4    1+I.INVP                                                GPO          3454     A
GPO                BX6    X4+X3       INV[LIP] = 1                                GPO          3455     A
GPO                LX3    -I.IICP                                                 GPO          3456     A
GPO                SB2    X3          AC = IIC[LIP]                               GPO          3457     A
GPO                SA6    A3                                                      GPO          3458     A
GPO                NZ     B2,RJ31     IF AC " 0                                   GPO          3459     A
GPO                                                                               GPO          3460     A
GPO       *        MARK RJ3 AND DEFS FOLLOWING AS *INV*                            GPO          3461     A
GPO                                                                               GPO          3462     A
GPO        RJ32    SX3    A7-"TB"     IIC[R1+3] = LII - TB                        GPO          3463     A
GPO                LX3    I.IICP                                                  GPO          3464     A
GPO                BX7    X4+X3       INV[R1+3] = 1                               GPO          3465     A
GPO                SA7    A5+B3       LII = R1 + 3                                GPO          3466     A
GPO                SA5    A7+B1       R1 = LII + 1                                GPO          3467     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                 UX1    B2,X5                                                GPO        3468     A
GPO                 SX2    B2-OC.DEF                                            GPO        3469     A
GPO                 ZR     X2,RJ32     IF OC[R1] = OC.DEF                       GPO        3470     A
GPO                 SA5    A5+4                                                 CCG0218       5     A
GPO                 UX1    B2,X5                                                CCG0218       6     A
GPO                 SX2    B2-OC.DEF                                            CCG0218       7     A
GPO                 ZR     X2,RJ32A    IF OC[R1] = OC.DEF                       CCG0218       8     A
GPO                 SA5    A5-4                            */ BACK UP           CCG0218       9     A
GPO                 UX1    B2,X5                                                CCG0218      10     A
GPO                 EQ     RJ32B                                                CCG0218      11     A
GPO                                                                             CCG0218      12     A
GPO       RJ32A     SX3    A7-"TB"     IIC[R1+3] = LII - TB                     CCG0218      13     A
GPO                 LX3    I.IICP                                               CCG0218      14     A
GPO                 BX7    X4+X3       INV[R1+3] = 1                            CCG0218      15     A
GPO                 SA7    A5+B3       LII = R1 + 3                             CCG0218      16     A
GPO                 SA5    A7+B1       R1 = LII + 1                             CCG0218      17     A
GPO                 UX1    B2,X5                                                CCG0218      18     A
GPO                                                                             GPO        3471     A
GPO                 JP     MII.JT+B2   JUMP( MII.JT( OC[R1] ) )                 GPO        3472          I
          -FCC1254
GPO        TST      SPACE  3,14                                                 GPO        3473          I
          -FCC1254
GPO       *         TST - DEFINITION POINT OF MOVED INSTRUCTION                 GPO        3474          I
          -FCC1254
GPO                                                                             GPO        3475          I
          -FCC1254
GPO                 PROCESS TST                                                 GPO        3476          I
          -FCC1254
GPO                 BX6    "RN"X5                                               GPO        3477          I
          -FCC1254
GPO                 SA4    "TB"+X6     LI = TB + RI[R1]                         GPO        3478          I
          -FCC1254
GPO                 SA3    O.TET                                                GPO        3479          I
          -FCC1254
GPO                 LX4    59-I.INVP                                            GPO        3480          I
          -FCC1254
GPO                 PL     X4,MII2     IF ^INV[LI]                              GPO        3481          I
          -FCC1254
GPO                                                                             GPO        3482          I
          -FCC1254
GPO                 SA1    A5+B1       R2 = R1 + 1                              GPO        3483          I
          -FCC1254
GPO                 SB2    X3                                                   GPO        3484          I
          -FCC1254
GPO                 LX1    -IH.CAP                                              GPO        3485          I
          -FCC1254
GPO                 SA2    B2+X1                                                GPO        3486          I
          -FCC1254
GPO                 LX2    59-T.INVP                                            GPO        3487          I
          -FCC1254
GPO                 PL     X2,MII2     IF ^INV[TET(CA[R2])]                     GPO        3488          I
          -FCC1254

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                EQ     MII1A                                                    GPO         3489          I
          -FCC1254
GPO                SA3    AN                                                       FCC1254       39          I
          -CCG0218
GPO       RJ32B    SA3    AN                                                       CCG0218       19     A
GPO                ZR     X3,RJ33     IF AN = 0     */ MII CALLED FROM *CIF*       FCC1254       40     A
GPO                SA3    XRF                                                      FCC1254       41     A
GPO                SX4    B1                                                       FCC1254       42     A
GPO                IX6    X3-X4       XRF = XRF - 1 */ DECR N.EXT-REFS IN LOOP     FCC1254       43     A
GPO                SA6    A3                                                       FCC1254       44     A
GPO                NZ     X6,RJ33     IF XRF " 0                                   FCC1254       45     A
GPO                                                                               FCC1254       46     A
GPO                SA3    MRA.X                                                    FCC1254       47     A
GPO                BX6    X3          MRA = MRA.X   */ MAKE REGS AVAIL IN *GRA*    FCC1254       48     A
GPO                SA6    MRA                                                      FCC1254       49     A
GPO                                                                               FCC1254       50     A
GPO       RJ33     JP     MII.JT+B2   JUMP( MII.JT( OC[R1] ) )                     FCC1254       51     A
GPO       ST       EJECT                                                           GPO         3490     A
GPO      *         ST - SEE IF MOVABLE OR A RECURSIVE DEFINITION                   GPO         3491     A
GPO                                                                               GPO         3492     A
GPO                PROCESS ST                                                      GPO         3493     A
GPO                SA3    A5+B1       R2 = R1 + 1                                  GPO         3494     A
GPO                BX6    "RN"X5                                                  GPO         3495     A
GPO                LX5    -R1.INP                                                 GPO         3496     A
GPO                SA4    "UB"+X5     UI = UDI(IN[R1])                            GPO         3497     A
GPO                LX3    -IH.RFP                                                 GPO         3498     A
GPO                SX7    X3                                                      GPO         3499     A
GPO                NZ     X7,MII2     IF RF[R2] " 0 */ LEAVE INDEXED STORES ALONE  GPO         3500     A
GPO                                                                               GPO         3501     A
GPO                SA2    "TB"+X6     LI = TB + RI[R1]                            GPO         3502     A
GPO                LX2    59-I.INVP                                               GPO         3503     A
GPO                SA1    AN                                                      GPO         3504     A
GPO                SB2    X5                                                      GPO         3505     A
GPO                BX7    X1*X2                                                   GPO         3506     A
GPO                ZR     B2,ST3      IF IN[R1] = 0 */ AP/IO LIST STORE            GPO         3507     A
GPO                                                                               GPO         3508     A
GPO                PL     X7,ST1      IF ^( AN & INV[LI] )                        GPO         3509     A
GPO                MX6    -UD.STCL                                                GPO         3510     A
GPO                LX6    UD.STCP+1                                               GPO         3511     A
GPO                BX1    -X6*X4                                                  GPO         3512     A
GPO                NZ     X1,ST1      IF ^( STC[UI] = 1 & IST[UI] = 0 )           GPO         3513     A
GPO                                                                               GPO         3514     A
GPO                SA3    A4+B1       U2 = UI + 1                                 GPO         3515     A
GPO                SA1    UBD                                                     GPO         3516     A
GPO                SA4    LEA                                                     FCCA513X      48     A
GPO                UX7    B2,X3       B = BITN[U2]                                GPO         3517     A
GPO                IX6    X1+X7                                                   GPO         3518     A
GPO                SA3    UVA                                                     GPO         3519     A
GPO                IX4    X4+X7                                                   FCCA513X      49     A
GPO                IX7    X3+X7                                                   GPO         3520     A
GPO                SA1    X6          UFW = UBD( WI[U2] )                         GPO         3521     A
GPO                SA4    X4          LEW = LEA( WI[U2] )                         FCCA513X      50     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                SA3    X7          PUW = UVA( WI[U2] )                          GPO         3522     A
GPO                BX6    X3+X1                                                    GPO         3523     A
GPO                BX6    X4+X6                                                    FCCA513X      51     A
GPO                AX7    B2,X6                                                    GPO         3524     A
GPO                LX7    59                                                       GPO         3525     A
GPO                MI     X7,ST1      IF SHIFT( PUW ! UFW , 59-B ) < 0  */ USED    GPO         3526          I
          -FCCA513X
GPO                SA4    "UB"+X5     RESET UI                                     FCCA513X      52     A
GPO                MI     X7,ST1      IF SHIFT(PUW!UFW!LEW!,59-B)                  FCCA513X      53     A
GPO                                                                               GPO         3527     A
GPO      *         *ST* IS MOVABLE, CLEAR BITS IN *UDT* AND BLOCK *DEF* VECTOR    GPO         3528     A
GPO      *         SET *RF* BIT IN IN DESCR TO TELL *RDD* NOT TO KILL IT.         GPO         3529          I
          -FCC1556X
GPO      *         SET BIT IN REGION LIVE ENTRY BIT VECTOR FOR *RDD* AND *GRA*    FCC1556X      12     A
GPO                                                                               GPO         3530     A
GPO                SA3    VL                                                       GPO         3531          I
          -FCC0408
GPO                MX7    1                                                        GPO         3532          I
          -FCC0408
GPO                LX7    1+UD.DEFRP                                               GPO         3533          I
          -FCC0408
GPO                BX6    -X7*X4      DEFR[UI] = 0  */ SO *LD*"S CAN MOVE          GPO         3534          I
          -FCC0408
GPO                SA3    STA                                                      FCC0408       87     A
GPO                BX6    -X3*X4      (DEFR,^RD,STC)[UI] = 0  */ SO LD"S CAN MOVE  FCC0408       88     A
GPO                SA3    VL                                                       FCC0408       89     A
GPO                SX7    B1                                                       GPO         3535     A
GPO                SA6    A4                                                       GPO         3536     A
GPO                LX7    B2,X7                                                    GPO         3537     A
GPO                SB2    X3                                                       GPO         3538     A
GPO                SA1    A1-B2       DEFW = UFW - VL                              GPO         3539     A
GPO                BX6    -X7*X1      DEFW = DEFW & ^SHIFT(1,B)                    GPO         3540     A
GPO                SA6    A1                                                       GPO         3541     A
GPO                SA4    A5+2        DI = R1 + 2                                  GPO         3542          I
          -FCC1556X
GPO                SX7    B1                                                       GPO         3543          I
          -FCC1556X
GPO                LX7    D.RFP                                                    GPO         3544          I
          -FCC1556X
GPO                BX6    X7+X4       RF[DI] = 1                                   GPO         3545          I
          -FCC1556X
GPO                SA6    A4                                                       GPO         3546          I
          -FCC1556X
GPO                SA3    LEA                                                      FCC1556X      13     A
GPO                SA4    A4+B1       U2 = UI + 1                                  FCC1556X      14     A
GPO                IX6    X3+X4                                                    FCC1556X      15     A
GPO                SA4    X6          LEW = [LEA+WI[U2]]                           FCC1556X      16     A
GPO                BX6    X7+X4       LEW = LEW ! SHIFT(1,B)  */ MOVED ST IS *LE*  FCC1556X      17     A
GPO                SA6    A4                                                       FCC1556X      18     A
GPO                EQ     MII1                                                     GPO         3547     A
GPO       ST       SPACE  1,10                                                     GPO         3548     A
GPO      *         CHECK TO SEE IF THIS *ST* IS A RECURSIVE DEFINITION            GPO         3549     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO      *         THE RESTRICTION THAT THE *ST* IMMEDIATELY FOLLOW THE INCREMENT        GPO        3550     A
GPO      *         IS A SIMPLE WAY OF DISALLOWING USES OF THE INC BETWEEN IT AND         GPO        3551     A
GPO      *         THE *ST*. WITHOUT THIS WE COULD GET INTO SOME NASTY SITUATIONS        GPO        3552     A
GPO      *         AS THE RESULT OF REGISTER ASSIGNMENT.                                 GPO        3553     A
GPO                                                                                      GPO        3554     A
GPO       ST1      LX4    59-UD.^RDP                                                     GPO        3555     A
GPO                MI     X4,MII2     IF ^RD[UI]                                         GPO        3556     A
GPO                LX2    I.INVP-I.IAP                                                   GPO        3557     A
GPO                PL     X2,ST2      IF ^IA[LI]    */ PRED IS NOT  I+INV                GPO        3558     A
GPO                                                                                      GPO        3559     A
GPO                LX2    1+I.IAP-I.LDIP                                                 GPO        3560     A
GPO                SX6    "SC"                                                           GPO        3561     A
GPO                BX1    X5-X2                                                          GPO        3562     A
GPO                SB2    X1                                                             GPO        3563     A
GPO                NZ     B2,ST2      IF LDI[LI] " IN[R1]  */ NOT  I = I+INV             GPO        3564     A
GPO                SB2    A2+B1                                                          GPO        3565     A
GPO                SX1    A5-B2                                                          GPO        3566     A
GPO                NZ     X1,ST2      IF LI+1 " R1  */ INC OP NOT NEXT TO *ST*           GPO        3567     A
GPO                                                                                      GPO        3568     A
GPO      *         ADD *ST* TO *STC*, LIST OF POSSIBLE *RD*"S                            GPO        3569     A
GPO                                                                                      GPO        3570     A
GPO                LX6    I.STCP      STC[R1+3] = SC                                     GPO        3571     A
GPO                SA6    A5+B3                                                          GPO        3572     A
GPO                S"NI"  "NI"+B1     NI = NI + 1                                        GPO        3573     A
GPO                S"SC"  A6-"TB"     SC = R1+3 - TB                                     GPO        3574     A
GPO                EQ     MII2                                                           GPO        3575     A
GPO                                                                                      GPO        3576     A
GPO      *         MARK VAR AS *^RD* IN *UDT*                                            GPO        3577     A
GPO                                                                                      GPO        3578     A
GPO       ST2      MX1    1                                                              GPO        3579     A
GPO                BX6    X1+X4                                                          GPO        3580     A
GPO                LX6    1+UD.^RDP   ^RD[UI] = 1                                        GPO        3581     A
GPO                SA6    A4                                                             GPO        3582     A
GPO                EQ     MII2                                                           GPO        3583     A
GPO                                                                                      GPO        3584     A
GPO      *         AP / IO LIST STORE, MOVABLE IF ^EQUIVALENCED                          GPO        3585     A
GPO                                                                                      GPO        3586     A
GPO       ST3      PL     X7,MII2     IF ^( AN & INV[R1] )                               GPO        3587     A
GPO       .AP      IF     DEF,I.AP                                                       CCG7        109     A
GPO                LX3    IH.RFP-IH.IHP                                                  GPO        3588     A
GPO                SX7    X3-I.IO                                                        GPO        3589          I
          -CCG18
GPO                PL     X7,MII1     IF I[R2] = I.IO  */ AN I/O LIST                    GPO        3590          I
          -CCG18
GPO                SA1    =XO.API                                                        GPO        3591          I
          -CCG18
GPO                SB2    X3-I.AP-1                                                      GPO        3592          I
          -CCG18
GPO                SA2    X1+B2                                                          GPO        3593          I
          -FCC0672 -CCG18
GPO       *        LX2    59-AI.EQVP                                                     GPO        3594          I
          -FCC0672 -CCG18

          0        1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                MI     X2,MII2     IF EQV[API(H[R2])]                          GPO         3595          I
          -FCC0672  -CCG18
GPO                SA2    X1+B2       AI = API(H[R2])                             FCC0672        7          I
          -CCG18
GPO                SX7    X3                                                      CCG18         93     A
GPO                AX7    IH.IP                                                   CCG18         94     A
GPO                SA1    =XO$APT-1+X7                                            CCG18         95          I
          -CCGA030
GPO                SA1    =XO$APT-2+X7
>>>>      CCGA030 11                                                              CCGA030        5     A
GPO                MX7    -IH.HL                                                  CCG18         96     A
GPO                BX3    -X7*X3                                                  CCG18         97     A
GPO                SB2    X3                                                      CCG18         98     A
GPO                SA2    X1+B2       AI = APT(H[R2])                             CCG18         99     A
GPO                MX7    2                                                       FCC0672        8     A
GPO                BX3    X7*X2                                                   FCC0672        9     A
GPO                NZ     X3,MII2     IF EQV[AI] ! BASE[AI]                       FCC0672       10          I
          -CCG7
GPO                EQ     MII1                                                    GPO         3596          I
          -CCG7
GPO                ZR     X3,MII1     IF ^( EQV[AI] ! BASE[AI] )                  CCG7         110     A
GPO        .AP     ENDIF                                                          CCG7         111     A
GPO                EQ     MII2                                                    CCG7         112     A
GPO                                                                               FCC0408       90     A
GPO        STA     BFMW   UD,(DEFR,^RD,STC)                                       FCC0408       91     A
GPO        EOQ     EJECT                                                          GPO         3597     A
GPO       *        EOQ - END OF MARKING PHASE, SET CHAIN POINTERS IN *BSW*        GPO         3598     A
GPO                                                                               GPO         3599     A
GPO                PROCESS EOQ                                                    GPO         3600     A
GPO                SA1    BN                                                      GPO         3601     A
GPO                SX5    "SC"                                                    GPO         3602     A
GPO                SX7    A7-"TB"     IIC = LII - TB                              GPO         3603     A
GPO                LX5    BS.STCP     STC[BSW] = SC                               GPO         3604     A
GPO                LX7    BS.IICP     IIC[BSW] = LII - TB                         GPO         3605     A
GPO                BX6    X5+X7                                                   GPO         3606     A
GPO                PX7    "NI",X6     NIS[BSW] = NI                               GPO         3607     A
GPO                SA7    "TB"        [TB] = BSW                                  GPO         3608     A
GPO                SA7    BSW                                                     GPO         3609     A
GPO                ZR     X1,MII      IF BN = 0     */ CALL FROM *CIF*            GPO         3610     A
GPO                TRACE  MII,RLIST,SEQ                                           GPO         3611     A
GPO                                                                               GPO         3612     A
GPO       *        ADD CONTRIBUTIONS FROM THIS BLOCK INTO REGION USE/DEF VECTORS. GPO         3613     A
GPO                                                                               GPO         3614     A
GPO                SA1    UBD                                                     GPO         3615     A
GPO                SA2    VL                                                      GPO         3616     A
GPO                SA3    DVA                                                     GPO         3617     A
GPO                SB2    B0          I = 0                                       GPO         3618     A
GPO                SB3    X2                        (B3) = VL                     GPO         3619     A
GPO                SB4    B3+B3                                                   GPO         3620     A
GPO                IX0    X1-X2       BDA = UBD - VL                              GPO         3621     A
GPO                SB6    X3                        (B6) = [DVA]                  GPO         3622     A
GPO                                                                               GPO         3623     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO       MII4     SA1     X0+B2                                                GPO        3624     A
GPO                SA2     B6+B2                                                GPO        3625     A
GPO                SA3     A1+B4                                                GPO        3626     A
GPO                SA4     A2+B4                                                GPO        3627     A
GPO                BX6     X1+X2       DVA(I) = DVA(I) ! BDA(I)                 GPO        3628     A
GPO                SA6     A2                                                   GPO        3629     A
GPO                BX7     X3+X4       UVA(I) = UVA(I) ! BUA(I)                 GPO        3630     A
GPO                SB2     B2+B1       I = I + 1                                GPO        3631     A
GPO                SA7     A4                                                   GPO        3632     A
GPO                LT      B2,B3,MII4  IF I < VL                                GPO        3633     A
GPO                                                                             GPO        3634     A
GPO                ZR      "TB",MII5   IF TB = 0                                GPO        3635     A
GPO                SA5     O.BIT                                                GPO        3636     A
GPO                SA4     "TB"-3      R1 = TB - 3                              GPO        3637     A
GPO                SB2     X5                                                   GPO        3638     A
GPO                SA3     B2+X4       BIW = BIT(BN[R1])                        GPO        3639     A
GPO                LX3     59-BI.RBP                                            GPO        3640     A
GPO                MI      X3,MII5     IF RB[BIW]    */ NOT A HOLDING BLOCK     GPO        3641     A
GPO                                                                             GPO        3642     A
GPO                LX3     1+BI.RBP-BI.PBNP                                     GPO        3643     A
GPO                SA2     B2+X3       BIW = BIT(PBN[BIW])                      GPO        3644     A
GPO                SA1     O.BVT                                                GPO        3645     A
GPO                LX2     -BI.BVIP                                             GPO        3646     A
GPO                IX0     X1+X2       BDA = O.BVT + BVI[BIW]                   GPO        3647     A
GPO                SB2     B0          I = 0                                    GPO        3648     A
GPO                S"TB"   B0          TB = 0                                   GPO        3649     A
GPO                EQ      MII4                                                 GPO        3650     A
GPO                                                                             GPO        3651     A
GPO       MII5     SA2     IJP                                                  GPO        3652     A
GPO                ZR      X2,MII      IF IJP = 0    */ NO INVARIANT JUMPS      GPO        3653     A
GPO                                                                             GPO        3654     A
GPO                EQ      MII                                                  GPO        3655     A
GPO       MII.JT   SPACE   3,14                                                 GPO        3656     A
GPO      *         THE FOLLOWING OPCODES ARE NOT EXPECTED TO APPEAR             GPO        3657     A
GPO                                                                             GPO        3658     A
GPO                ECHO    1,X=(PLD,PST,SLD,SST,SLD,SDS,DRL,DWL,TLD)            GPO        3659          I
         -FCCA506X
GPO                ECHO    1,X=(PLD,PST,SLD,SST,SDS,DRL,DWL)                    FCCA506X      5     A
GPO       .X       EQU     1S17+MII                                             GPO        3660     A
GPO                                                                             GPO        3661     A
GPO      *         OPR - DEFINE MACRO TO FORM JUMP TABLE FOR *MII*              GPO        3662     A
GPO                                                                             GPO        3663     A
GPO                MACRO   OPR,NAM,SS                                           GPO        3664     A
GPO                NOREF   .NAM                                                 GPO        3665     A
GPO       O        IF      DEF,.NAM                                             GPO        3666     A
GPO       Z        MICRO   3,1,/NAM/                                            GPO        3667     A
GPO                IFC     NE,/"Z"//,2                                          GPO        3668     A
GPO                EQ      .NAM              NAM                                GPO        3669     A
GPO                SKIP    1                                                    GPO        3670     A
GPO                EQ      .NAM               NAM                               GPO        3671     A
GPO       O        ELSE    2                                                    GPO        3672     A
GPO       TYP      MICRO   3,3,$SS$                                             GPO        3673     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                EQ     TY"TYP"     NAM                                    GPO        3674     A
GPO                ENDM                                                      GPO        3675     A
GPO                SPACE  2                                                  GPO        3676     A
GPO       MII.JT   BSS    0                                                  GPO        3677     A
GPO                LOC    0                                                  GPO        3678     A
GPO      *CALL     OPRDEFS                                                   GPO        3679     A
GPO                                                                          GPO        3680     A
GPO                LOC    *O                                                 GPO        3681     A
GPO                LIST   -G                                                 GPO        3682     A
GPO                                                                          GPO        3683     A
GPO                QUAL   CODE                                               GPO        3684     A
GPO                                                                          GPO        3685     A
GPO       MII      EQU    /MII/MII                                           GPO        3686     A
GPO       FII      TITLE  FII - FORM INCREMENT INFORMATION                   GPO        3687     A
GPO      **        FII - FORM INCREMENT INFORMATION                          GPO        3688     A
GPO      *         SCAN *STC* TO FORM A LIST OF INCREMENT VALUES OF *RD*     GPO        3689     A
GPO      *         VARIABLES, AND SAVE THEM IN *IIT*.                        GPO        3690     A
GPO      *                                                                   GPO        3691     A
GPO      *         EXIT   *STC* IN BSW UPDATED                               GPO        3692     A
GPO      *                *HBP* OF *ST* OF *RD* POINTS TO INCREMENT VALUE.   GPO        3693     A
GPO                                                                          GPO        3694     A
GPO       FII0     SA4    IIL                                                GPO        3695     A
GPO                SA5    BSW                                                GPO        3696     A
GPO                SX6    B5                                                 GPO        3697     A
GPO                LX0    BS.STCP                                            GPO        3698     A
GPO                SX3    B4                                                 GPO        3699     A
GPO                BX5    X0*X5                                              GPO        3700     A
GPO                SA6    L.IIT       L.IIT = IL    */ FINAL LENGTH          GPO        3701     A
GPO                LX3    BS.STCP     STC[BSW] = SC                          GPO        3702     A
GPO                BX1    X3+X5                                              GPO        3703     A
GPO                PX7    X1          NIS[BSW] = 0                           GPO        3704     A
GPO                SA7    A5                                                 GPO        3705     A
GPO                                                                          GPO        3706     A
GPO                TRACE  FII,IIT                                            GPO        3707     A
GPO                                                                          GPO        3708     A
GPO       FII      ROUTINE                                                   GPO        3709     A
GPO                UX6    B2,X4                                              GPO        3710     A
GPO                ALLOC  IIT,B2      ALLOC( IIT , NIS[BSW] )                GPO        3711     A
GPO                SA5    O.SEQ                                              GPO        3712     A
GPO                SA4    BSW                                                GPO        3713     A
GPO                SA3    O.UDT                                              GPO        3714     A
GPO                S"TB"  X5+3        TB = O.SEQ + 3                         GPO        3715     A
GPO                SX6    B6          IIL = OLD(L.IIT)                       GPO        3716     A
GPO                SB5    B6          IL = IIL      */ *IIT* LENGTH          GPO        3717     A
GPO                SA6    IIL                                                GPO        3718     A
GPO                SA0    X2                                                 GPO        3719     A
GPO                S"UB"  X3          UB = O.UDT                             GPO        3720     A
GPO                LX4    -BS.STCP                                           GPO        3721     A
GPO                SB3    X4          I = STC[BSW]                           GPO        3722     A
GPO                SB4    B0          SC = 0        */ LAST ON CHAIN         GPO        3723     A
GPO                EQ     FII2                                               GPO        3724     A
GPO                                                                          GPO        3725     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO      *         SAVE INC VALUE IN *IIT*, SET *ST* TO POINT TO *IIT*                GPO        3726     A
GPO                                                                                  GPO        3727     A
GPO       FII1     SA6    A0+B5       IIT(IL) = ITW                                  GPO        3728     A
GPO                SX4    B4                                                         GPO        3729     A
GPO                SX1    B5                                                         GPO        3730     A
GPO                LX4    I.STCP      STC[LI] = SC  */ POINT TO LAST                 GPO        3731     A
GPO                LX1    I.HBPP      HBP[LI] = IL  */ POINT TO INC VALUE            GPO        3732     A
GPO                PX6    X1          NPI[LI] = 0                                    GPO        3733     A
GPO                BX7    X4+X6                                                      GPO        3734     A
GPO                SB5    B5+B1       IL = IL + 1                                    GPO        3735     A
GPO                SA7    A5                                                         GPO        3736     A
GPO                SB4    A5-"TB"     SC = LI - TB  */ CHAIN POINTS TO THIS          GPO        3737     A
GPO                                                                                  GPO        3738     A
GPO       FII2     SA5    "TB"+B3     LI = TB + I                                    GPO        3739     A
GPO                ZR     B3,FII0     IF I = 0                                       GPO        3740     A
GPO                LX5    -I.STCP                                                    GPO        3741     A
GPO                SA4    A5-3        R1 = LI - 3                                    GPO        3742     A
GPO                SB3    X5          I = STC[LI]   */ NEXT ON CHAIN                 GPO        3743     A
GPO                BX6    "RN"X4      R = RI[R1]                                     GPO        3744     A
GPO                LX4    -R1.INP                                                    GPO        3745     A
GPO                SA3    "UB"+X4     UI = UB + IN[R1]                               GPO        3746     A
GPO                LX3    59-UD.^RDP                                                 GPO        3747     A
GPO                SA2    A5-B1       DI = LI - 1                                    GPO        3748     A
GPO                MI     X3,FII2     IF ^RD[UI]    */ ^ RD IN REGION                GPO        3749     A
GPO                                                                                  GPO        3750     A
GPO      *         *ST* IS STILL A *RD*, FIND THE INCREMENT VALUE                     GPO        3751     A
GPO                                                                                  GPO        3752     A
GPO                MI     X3,FII2     IF ^RD[UI]                                     GPO        3753          I
         -CCG36
GPO                SB2    "TB"-B1                                                    GPO        3754     A
GPO                SA3    B2+X6       DIP = TB-1 + R                                 GPO        3755     A
GPO                SA4    A3-2        R1 = DIP - 2                                   GPO        3756     A
GPO                SX1    B1                                                         GPO        3757     A
GPO                LX1    D.INCP                                                     GPO        3758     A
GPO                BX7    X1+X3       INC[DIP] = 1                                   GPO        3759     A
GPO                SA7    A3                                                         GPO        3760     A
GPO                BX6    X1+X2       INC[DI] = 1                                    GPO        3761     A
GPO                UX5    B2,X4                                                      GPO        3762     A
GPO                SA6    A2                                                         GPO        3763     A
GPO                SX7    B2-OC.STT                                                  GPO        3764     A
GPO                NZ     X7,FII3     IF OC[R1] " OC.STT                             GPO        3765     A
GPO                                                                                  GPO        3766     A
GPO                SA3    A4+B1       R2 = R1 + 1                                    GPO        3767     A
GPO                MX1    -IH.CAL                                                    GPO        3768     A
GPO                LX3    -IH.CAP                                                    GPO        3769     A
GPO                BX4    -X1*X3                                                     GPO        3770     A
GPO                PX6    X4          ITW = PACK(0,CA[R2])                           GPO        3771     A
GPO                EQ     FII1                                                       GPO        3772     A
GPO                                                                                  GPO        3773     A
GPO      *         PROCESS IA, IS                                                     GPO        3774     A
GPO                                                                                  GPO        3775     A
GPO       FII3     LX4    -R1.RJP                                                    GPO        3776     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                BX6    "RN"X4                                                    GPO        3777    A
GPO                SA3    "TB"+X6     LP = TB + RJ[R1]                              GPO        3778    A
GPO                MX5    0           BIAS = 0                                      GPO        3779    A
GPO                SX7    A3          LA = LP                                       GPO        3780    A
GPO                LX3    59-I.INVP                                                 GPO        3781    A
GPO                MI     X3,FII4     IF INV[LP]                                    GPO        3782    A
GPO                LX4    R1.RJP-R1.RKP                                             GPO        3783    A
GPO                BX6    "RN"X4                                                    GPO        3784    A
GPO                SX5    B2-OC.IA    BIAS = OC[R1] - OC.IA                         GPO        3785    A
GPO                SX7    "TB"+X6     LA = TB + RK[R1]                              GPO        3786    A
GPO                                                                                GPO        3787    A
GPO      *         INC VALUE IS +- VAL OF *INV* PRED OF IA/IS                        GPO        3788    A
GPO                                                                                GPO        3789    A
GPO       FII4     SA4    X7-3        R1 = LA - 3                                   GPO        3790    A
GPO                UX6    B2,X4                                                     GPO        3791    A
GPO                SX1    B2-OC.LD                                                  GPO        3792    A
GPO                NZ     X1,FII5     IF OC[R1] " OC.LD                             GPO        3793    A
GPO                                                                                GPO        3794    A
GPO                SA4    A4+B1       R2 = R1 + 1                                   GPO        3795    A
GPO                AX6    R1.INP                                                    GPO        3796    A
GPO                SB2    X5+B1                                                     GPO        3797    A
GPO                LX6    IH.RFP                                                    GPO        3798    A
GPO                BX5    X6+X4       VAL = SHIFT(IN[R1],IH.RFP) + CAIH[R2]         GPO        3799    A
GPO                PX6    B2,X5       ITW = PACK( BIAS+1 , VAL )                    GPO        3800    A
GPO                EQ     FII1                                                      GPO        3801    A
GPO       FII      SPACE  3,14                                                      GPO        3802    A
GPO      *         OPERATION IS ZP , SHOULD BE A *S* OR *FMA*                        GPO        3803         I
         -CCG36
GPO      *         OPERATION IS ZP, SHOULD BE *S*, *FMA* OR *LDC*          000410    CCG36        24    A
GPO                                                                                GPO        3804    A
GPO       FII5     SX7    B2-OC.S                                                   GPO        3805    A
GPO                MX1    -R1.INL                                                   GPO        3806    A
GPO                LX4    -R1.INP     C = IN[R1]                                    GPO        3807    A
GPO                ZR     X7,FII6     IF OC[R1] = OC.S                              GPO        3808    A
GPO                SX7    B2-OC.FMA                                                 GPO        3809    A
GPO                NZ     X7,FII10    IF OC[R1] " OC.FMA                            GPO        3810    A
GPO                SB2    X4-1                                                      GPO        3811    A
GPO                MX2    1                                                         GPO        3812    A
GPO                AX4    B2,X2       C = SHIFT( 1S59 , -(C-1) )                    GPO        3813    A
GPO                                                                                GPO        3814    A
GPO       FII6     ZR     X5,FII7     IF BIAS = 0                                   GPO        3815    A
GPO                BX4    -X4         C = -C                                        GPO        3816    A
GPO       FII7     BX4    -X1*X4                                                    GPO        3817    A
GPO                PX6    X4          ITW = PACK( 0 , C )                           GPO        3818    A
GPO                EQ     FII1                                                      GPO        3819    A
GPO                                                                                GPO        3820    A
GPO       FII10    EQU    FII+1S17    COMPILER ERROR                                GPO        3821         I
         -CCG36
GPO       FII10    SX7    B2-OC.LDC                                       000430    CCG36        25    A
GPO                NZ     X7,*+1S17   IF OC[R1] " OC.LDC                      000440    CCG36        26    A
GPO                SA2    =XO$CVT                                         000450    CCG36        27    A
GPO                SA4    A4+B1       R2 = R1 + 1                         000460    CCG36        28    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                LX4    -IH.CAP                                          000470   CCG36        29     A
GPO                SB2    X2                                               000480   CCG36        30     A
GPO                SA4    B2+X4       C = CVT(CA[R2])                      000490   CCG36        31     A
GPO                EQ     FII6                                             000500   CCG36        32     A
GPO        MIP     TITLE  MIP - MARK INTEGER POLYNOMIALS                           GPO        3822     A
GPO       **       MIP - MARK INTEGER POLYNOMIALS AND MERGE ONTO CHAIN OF *INV*    GPO        3823     A
GPO       *        INSTRUCTIONS.                                                   GPO        3824     A
GPO       *        FOWARD SCAN TO FORM A LIST OF INTEGER POLYNOMIALS ( *IP*"S ).   GPO        3825     A
GPO       *        FOLLOWED BY A BACKWARDS SCAN OVER THE LIST TO MERGE THEM        GPO        3826     A
GPO       *        ONTO THE *IIC*.  DURING THE MARKING PHASE WE COMPUTE THE        GPO        3827     A
GPO       *        *COST* OF AN *IP*, WHICH IS AN APPROXIMATE MEASURE OF THE       GPO        3828     A
GPO       *        NUMBER OF TERMS IN IT.  THE DIFFERENCE BETWEEN THE COST OF      GPO        3829     A
GPO       *        AN *IP* AND ITS INCREMENT FORMULA ( 2 OR 3 ) IS AN APPROXIMATE  GPO        3830     A
GPO       *        MEASURE OF THE PROFIT IN CREATING A COMPILER TEMPORARY .        GPO        3831     A
GPO                                                                               GPO        3832     A
GPO                QUAL   MIP                                                     GPO        3833     A
GPO                                                                               GPO        3834     A
GPO        MIP0    BSS    0                                                       GPO        3835     A
GPO                TRACE  MIP,RLIST,SEQ                                           GPO        3836     A
GPO                                                                               GPO        3837     A
GPO        MIP     ROUTINE                                                        GPO        3838     A
GPO                SA1    L.SEQ                                                  GPO        3839     A
GPO                AX1    2                                                      GPO        3840     A
GPO                ALLOC  RND,X1      ALLOC( RND , L.SEQ/4 )                     GPO        3841     A
GPO                SA3    O.UDT                                                  GPO        3842     A
GPO                SA4    O.SEQ                                                  GPO        3843     A
GPO                MX7    0                                                      GPO        3844     A
GPO                SB5    X2                                                     GPO        3845          I
          -FCC0717
GPO                SB5    B0          LEI = 0       */ LAST ENCOUNTERED *INC* ST FCC0717        8     A
GPO                SA7    L.RND       L.RND = 0                                  GPO        3846     A
GPO                SA7    X2          PI = O.RND;  [PI] = 0                      GPO        3847     A
GPO                SB4    59-D.INCP                                              GPO        3848     A
GPO                S"UB"  X3          UB = O.UDT                                 GPO        3849     A
GPO                SB3    3                                                      GPO        3850     A
GPO                MX1    -I.COSTL                                               GPO        3851     A
GPO                S"TB"  X4+B3       TB = O.SEQ + 3                             GPO        3852     A
GPO                LX1    I.COSTP                                                GPO        3853     A
GPO                SA5    X4          R1 = O.SEQ                                 GPO        3854     A
GPO                BX6    X5                                                     FCC2271X       5     A
GPO                LX6    -IM.APLNL-R1.H2P  ARTICULATION NODE TO SIGN BIT        FCC2271X       6     A
GPO                SA6    AN          MI IFF UNCONDITIONALLY EXECUTED (SAVED IN * FCC2271X       7     A
GPO                EQ     MIP2                                                   GPO        3855     A
GPO        MIP     SPACE  3                                                      GPO        3856     A
GPO       *        ADD INSTRUCTION TO *IP* LIST                                   GPO        3857          I
          -FCC0717
GPO       *        ADD INSTRUCTION TO *IP* LIST, (A6,X6) = LI, [LI]               FCC0717        9     A
GPO                                                                               GPO        3858     A
GPO        MIP1    LX4    59-I.INVP                                              GPO        3859     A
GPO                MI     X4,MIP2     IF INV[LI]                                 GPO        3860     A
GPO                SX7    A4-"TB"                                                GPO        3861     A
GPO                SA7    A7+B1       PI = PI + 1;  [PI] = LI - TB               GPO        3862     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

GPO                ZR     B5,MIP2     IF LEI = 0    */ NO INCREMENTS YET          FCC0717      10     A
GPO                BX7    X6                                                      FCC0717      11     A
GPO                LX7    59-I.LDIP                                               FCC0717      12     A
GPO                MI     X7,MIP2     IF ODD[LDI[LI]]  */ USES NEW VALUE          FCC0717      13     A
GPO                                                                               FCC0717      14     A
GPO      *         IP USES OLD VALUE OF INCREMENT VARIABLE, MARK IT AS *AIP*     FCC0717      15     A
GPO      *         IF IT OCCURS PAST A REDEF OF THE INCREMENT VARIABLE.          FCC0717      16     A
GPO                                                                               FCC0717      17     A
GPO                SA3    BSW                                                     FCC0717      18          I
          -FCC1532X
GPO                LX3    -BS.STCP                                                FCC0717      19          I
          -FCC1532X
GPO                SB2    X3          J = STC[BSW]  */ FIRST INC ST IN BLOCK      FCC0717      20          I
          -FCC1532X
GPO                SA3    SC                                                      FCC1532X      6     A
GPO                SB2    X3          J = SC        */ FIRST INC ST IN BLOCK      FCC1532X      7     A
GPO                LX6    -I.LDIP     L = LDI[LI]                                 FCC0717      21     A
GPO                                                                               FCC0717      22     A
GPO       MIP1A    SA4    "TB"+B2     LIS = TB + J  */ LINK WD OF INC ST          FCC0717      23     A
GPO                SA3    A4-B3       R1S = LIS - 3 */ R1 WORD                    FCC0717      24     A
GPO                LX3    -R1.INP                                                 FCC0717      25     A
GPO                BX5    X3-X6                                                   FCC0717      26     A
GPO                SX2    X5                                                      FCC0717      27     A
GPO                ZR     X2,MIP1B    IF IN[R1S] = L  */ SAME VARIABLE            FCC0717      28     A
GPO                LX4    -I.STCP                                                 FCC0717      29     A
GPO                SB2    X4+         J = STC[LIS]  */ NEXT ON CHAIN              FCC0717      30     A
GPO                ZR     B2,MIP2     IF J = 0      */ END OF CHAIN               FCC0717      31     A
GPO                LE     B2,B5,MIP1A IF J @ LEI                                  FCC0717      32     A
GPO                EQ     MIP2                                                    FCC0717      33     A
GPO                                                                               FCC0717      34     A
GPO       MIP1B    SX4    B1                                                      FCC0717      35     A
GPO                LX4    I.AIPP                                                  FCC0717      36     A
GPO                LX6    I.LDIP                                                  FCC0717      37     A
GPO                BX6    X4+X6       AIP[LI] = 1                                 FCC0717      38     A
GPO                SA6    A6                                                      FCC0717      39     A
GPO                                                                               GPO        3863     A
GPO      *         ADVANCE TO NEXT INSTRUCTION, JUMP TO PROCESSOR                GPO        3864     A
GPO                                                                               GPO        3865     A
GPO       MIP2     SA5    A5+4        R1 = R1 + 4                                 GPO        3866     A
GPO                SA4    A5+B3       LI = R1 + 3                                 GPO        3867     A
GPO                UX6    B2,X5                                                   GPO        3868     A
GPO                JP     MIP.JT+B2   JUMP( MIP.JT( OC[R1] ) )                    GPO        3869     A
GPO                                                                               GPO        3870     A
GPO       MIPA     BFMW   I,(IP,COST,LDI)                                         GPO        3871     A
GPO       MIPB     BFMW   I,(IIC,LDI)                                            GPO        3872     A
GPO       MIPC     BFMW   I,(IP,C1)                                              GPO        3873     A
GPO       MIPD     BFMW   I,(IP,LDI)                                             GPO        3874     A
GPO       EOQ      SPACE  3,20                                                   GPO        3875          I
          -FCC0717
GPO       ST       SPACE  3                                                      FCC0717      40     A
GPO      *         ST - SET *LEI* IF AN INCREMENT STORE                          FCC0717      41     A
GPO                                                                               FCC0717      42     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                 PROCESS ST                                                  FCC0717      43     A
GPO                 SA3     A4-B1       DI = LI - 1                             FCC0717      44          I
          -FCC1504X
GPO                 BX6     "RN"X5                                              FCC1504X     19     A
GPO                 LX5     -R1.INP                                             FCC1504X     20     A
GPO                 SA3     "UB"+X5     UI = UDT(IN[R1])                        FCC1504X     21     A
GPO                 LX3     59-UD.RFP                                           FCC1504X     22     A
GPO                 PL      X3,ST1      IF ^RF[UI]    */ NOT SHORT USE          FCC1504X     23     A
GPO                 SB2     "TB"-B1                                             FCC1504X     24     A
GPO                 SA3     B2+X6       DIP = TB-1 + RI[R1]                     FCC1504X     25     A
GPO                 SX6     B1                                                  FCC1504X     26     A
GPO                 LX6     D.RFP                                               FCC1504X     27     A
GPO                 BX6     X6+X3       RF[DIP] = 1   */ PROPAGATE *RF* TO ST PRED FCC1504X  28     A
GPO                 SA6     A3                                                  FCC1504X     29     A
GPO                                                                             FCC1504X     30     A
GPO       ST1       SA3     A4-B1       DI = LI - 1                             FCC1504X     31     A
GPO                 LX3     59-D.INCP                                           FCC0717      45     A
GPO                 PL      X3,MIP2     IF ^INC[DI]                             FCC0717      46     A
GPO                                                                             FCC0717      47     A
GPO                 SB5     A4-"TB"     LEI = LI - TB  */ TXT INDEX OF LAST INC ST FCC0717   48          I
          -FCC1532X
GPO                 NZ      B5,ST2      IF LEI " 0    */ ^ FIRST INC ST IN BLOCK FCC1532X     8     A
GPO                 SX6     A4-"TB"                                             FCC1532X     9     A
GPO                 SA6     SC          SC = LI - TB  */ TXT INDEX OF FIRST INC ST FCC1532X  10     A
GPO                                                                             FCC1532X     11     A
GPO       ST2       SB5     A4-"TB"     LEI = LI - TB */ LAST INC ST            FCC1532X     12     A
GPO                 EQ      MIP2                                                FCC0717      49     A
GPO       BBM       SPACE 2                                                     FCC0717      50     A
GPO       *         BASIC BLOCK MARKER, CLEAR *LEI*                             FCC0717      51     A
GPO                                                                             FCC0717      52     A
GPO                 PROCESS (RJ3,RJ6,LAB)                                       FCC0717      53     A
GPO                 SB5     B0          LEI = 0                                 FCC0717      54     A
GPO                 EQ      MIP2                                                FCC0717      55     A
GPO       EOQ       EJECT                                                       FCC0717      56     A
GPO       **        EOQ - MERGE *IP* LIST INTO *ICC* CHAIN                      GPO        3876     A
GPO                                                                             GPO        3877     A
GPO                 PROCESS EOQ                                                 GPO        3878     A
GPO                 SA2     O.RND                                               FCC0717      57     A
GPO                 SB5     X2+                                                 FCC0717      58     A
GPO                 SB4     A7-B5       NP = PI - O.RND                         GPO        3879     A
GPO                 SA5     A7          PI = PI                                 GPO        3880     A
GPO                 LE      B4,B1,MIP0  IF NP @ 1                               GPO        3881     A
GPO                                                                             GPO        3882     A
GPO                 SA2     BSW                                                 GPO        3883     A
GPO                 SA1     MIPB                                                GPO        3884     A
GPO                 LX2     -BS.IICP                                            GPO        3885     A
GPO                 SB6     I.IICP                                              GPO        3886     A
GPO                 SB2     X2          IC = IIC[BSW]                           GPO        3887     A
GPO                 SB3     X5          PL = [PI]                               GPO        3888     A
GPO                 GT      B3,B2,MIP3  IF PL > IC                              GPO        3889     A
GPO                 SB4     B2          LE = IC                                 GPO        3890     A
GPO                 SA4     "TB"+B2     LI = TB + IC                            GPO        3891     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO              AX7    B6,X4                                          GPO        3892    A
GPO              SB2    X7          IC = IIC[LI]                       GPO        3893    A
GPO              EQ     MIP4                                           GPO        3894    A
GPO                                                                    GPO        3895    A
GPO       MIP3   BX2    X0*X2                                          GPO        3896    A
GPO              BX6    X2+X5       IIC[BSW] = PL */ UPDATE CHAIN POINTER GPO     3897    A
GPO              LX6    BS.IICP                                        GPO        3898    A
GPO              SA6    A2+                                            GPO        3899    A
GPO              SB4    X5          LE = PL                            GPO        3900    A
GPO              SA5    A5-B1       PI = PI - 1                        GPO        3901    A
GPO              SB3    X5          PL = [PI]                          GPO        3902    A
GPO              SA4    "TB"+B4     LI = TB + LE                       GPO        3903    A
GPO                                                                    GPO        3904    A
GPO       MIP4   GT     B2,B3,MIP5  IF IC > PL                         GPO        3905    A
GPO              SX3    B3                                             GPO        3906    A
GPO              BX7    -X1*X4      HBP[LI] = 0                        GPO        3907    A
GPO              LX3    I.IICP                                         GPO        3908    A
GPO              BX6    X3+X7       IIC[LI] = PL                       GPO        3909    A
GPO              SA6    A4                                             GPO        3910    A
GPO              SB4    B3          LE = PL                            GPO        3911    A
GPO              SA5    A5-B1       PI = PI - 1                        GPO        3912    A
GPO              SA4    "TB"+B3     LI = TB + PL                       GPO        3913    A
GPO              SB3    X5+         PL = [PI]                          GPO        3914    A
GPO              NZ     X5,MIP4     IF PL " 0                          GPO        3915    A
GPO              ZR     B2,MIP6     IF IC = 0                          GPO        3916    A
GPO                                                                    GPO        3917    A
GPO       MIP5   SX3    B2                                             GPO        3918    A
GPO              BX7    -X1*X4      HBP[LI] = 0                        GPO        3919    A
GPO              LX3    I.IICP                                         GPO        3920    A
GPO              BX6    X3+X7       IIC[LI] = IC                       GPO        3921    A
GPO              SA6    A4                                             GPO        3922    A
GPO              SB4    B2          LE = IC                            GPO        3923    A
GPO              SA4    "TB"+B2     LI = TB + IC                       GPO        3924    A
GPO              AX7    B6,X4                                          GPO        3925    A
GPO              SB2    X7          IC = IIC[LI]                       GPO        3926    A
GPO              GT     B2,B3,MIP5  IF IC > PL                         GPO        3927    A
GPO              NZ     B3,MIP4     IF PL " 0                          GPO        3928    A
GPO                                                                    GPO        3929    A
GPO        MIP6  BX6    -X1*X4      IIC[LI] = HBP[LI] = 0              GPO        3930    A
GPO              SA6    A4                                             GPO        3931    A
GPO              EQ     MIP0                                           GPO        3932    A
GPO        TYII  SPACE  3,8                                            GPO        3933    A
GPO       *      TYPE II - SET *IP* BIT                                GPO        3934    A
GPO                                                                    GPO        3935    A
GPO              PROCESS (FMA,S)                                       GPO        3936    A
GPO              SX2    B1                                             GPO        3937    A
GPO              LX2    I.IPP                                          GPO        3938    A
GPO              BX6    X2+X4       IP[LI] = 1                         GPO        3939    A
GPO              SA6    A4                                             GPO        3940    A
GPO              EQ     MIP2                                           GPO        3941    A
GPO        LD    SPACE  3,14                                           GPO        3942    A
GPO       *      LD - IF *RD* , THEN SET *IP* BIT , AND *LDI* FIELD IF ^INV GPO   3943    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                                                                    GPO        3944     A
GPO                 PROCESS (LD,ILD)                                   GPO        3945          I
          -CCG16
GPO                 PROCESS (LD,LDC,LDV)   ILD -> LD IN MII            CCG16       204     A
GPO                 SA3    A5+B1       R2 = R1 + 1                     GPO        3946     A
GPO                 LX5    -R1.INP                                     GPO        3947     A
GPO                 SA2    "UB"+X5     UI = UB + IN[R1]                GPO        3948     A
GPO                 LX3    -IH.RFP                                     GPO        3949     A
GPO                 BX6    "RN"X3                                      GPO        3950     A
GPO                 NZ     X6,MIP2     IF RF[R2] " 0                   GPO        3951     A
GPO                 LX2    59-UD.^RDP                                  GPO        3952     A
GPO                 MI     X2,MIP2     IF ^RD[UI]                      GPO        3953     A
GPO                                                                    GPO        3954     A
GPO                 SA3    MIPC                                        GPO        3955     A
GPO                 BX6    X3+X4       IP[LI] = 1;  COST[LI] = 1       GPO        3956     A
GPO                 LX4    59-I.INVP                                   GPO        3957     A
GPO                 MI     X4,LD1      IF INV[LI]                      GPO        3958     A
GPO                                                                    GPO        3959     A
GPO                 SX7    A4-"TB"                                     GPO        3960     A
GPO                 SX5    X5                                          GPO        3961     A
GPO                 SA7    A7+B1       PI = PI + 1;  [PI] = LI - TB    GPO        3962     A
GPO                 LX5    I.LDIP      LDI[LI] = IN[R1]                GPO        3963     A
GPO                 BX6    X6+X5                                       GPO        3964     A
GPO       LD1       SA6    A4                                          GPO        3965     A
GPO                 EQ     MIP2                                        GPO        3966     A
GPO       XMTC      SPACE  2,8                                         GPO        3967     A
GPO                 PROCESS XMTC                                       GPO        3968     A
GPO                 LX5    -R1.RJP                                     GPO        3969          I
          -CCGA046
GPO                 SA2    MIPA                                        GPO        3970          I
          -CCGA046
GPO                 BX6    "RN"X5                                      GPO        3971          I
          -CCGA046
GPO                 SA3    "TB"+X6     LJ = TB + RJ[R1]                GPO        3972          I
          -CCGA046
GPO                 SX7    B1                                          GPO        3973          I
          -CCGA046
GPO                 BX6    X2*X3                                       GPO        3974          I
          -CCGA046
GPO                 LX3    59-I.IPP                                    GPO        3975          I
          -CCGA046
GPO                 PL     X3,MIP2     IF ^IP[LJ]                      GPO        3976          I
          -CCGA046
GPO                                                                    GPO        3977          I
          -CCGA046
GPO                 BX6    X6+X4       (IP,COST,LDI)[LI] = (IP,COST,LDI)[LJ]  GPO   3978          I
          -CCGA046
GPO                 LX7    I.COSTP                                     GPO        3979          I
          -CCGA046
GPO                 IX6    X7+X6       COST[LI] = COST[LI] + 1         GPO        3980          I
          -CCGA046
GPO                 SA6    A4                                          GPO        3981          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
          -CCGA046
GPO                EQ     MIP1                                                GPO         3982          I
          -CCGA046
GPO                EQ     MIP2                                                CCGA046       12     A
GPO        SXT     EJECT                                                      GPO         3983     A
GPO       *        SXT - CONSTANT SHIFT PREDECESSOR                           GPO         3984     A
GPO                                                                           GPO         3985     A
GPO                PROCESS SXT                                                GPO         3986     A
GPO                LX5    -R1.RJP                                             GPO         3987     A
GPO                SA2    MIPA                                                GPO         3988     A
GPO                BX7    "RN"X5                                              GPO         3989     A
GPO                SA3    "TB"+X7     LJ = TB + RJ[R1]                        GPO         3990     A
GPO                SA5    A4+B1       R1 = LI + 1   */ ADVANCE TO CON SHIFT   GPO         3991     A
GPO                BX6    X2*X3                                               GPO         3992     A
GPO                LX3    59-I.IPP                                            GPO         3993     A
GPO                PL     X3,MIP2     IF ^IP[LJ]                              GPO         3994     A
GPO                                                                           GPO         3995     A
GPO                UX2    B2,X5                                               GPO         3996     A
GPO                SA3    A5+B1       R2 = R1 + 1                             GPO         3997     A
GPO                SB2    B2-OC.KLS                                           GPO         3998     A
GPO                NZ     B2,MIP2     IF OC[R1] " OC.KLS                      GPO         3999     A
GPO                LX3    -IH.CAP                                             GPO         4000     A
GPO                SB2    X3-17                                               GPO         4001          I
          -CCG21
GPO                SB2    X3-24                                               CCG21         16     A
GPO                SX7    A4-"TB"                                             GPO         4002     A
GPO                PL     B2,MIP2     IF CA[R2] \ 17  */ SHIFT COUNT TOO BIG  GPO         4003          I
          -CCG21
GPO                PL     B2,MIP2     IF CA[R2] \ 24  */ SHIFT COUNT TOO BIG  CCG21         17     A
GPO                                                                           GPO         4004     A
GPO                LX4    59-I.INVP                                           GPO         4005     A
GPO                MI     X4,SXT1     IF INV[LI]                              GPO         4006     A
GPO                SA7    A7+B1       PI = PI + 1;  [PI] = LI - TB            GPO         4007     A
GPO                SX7    X7+4                                                GPO         4008          I
          -FCC0717
GPO                SA7    A7+B1       PI = PI + 1;  [PI] = LI+4 - TB          GPO         4009          I
          -FCC0717
GPO                                                                           GPO         4010     A
GPO        SXT1    SA4    A5+B3       LI = R1 + 3                             GPO         4011     A
GPO                SX7    B1                                                  GPO         4012     A
GPO                BX6    X6+X4       (IP,COST,LDI)[LI] = (IP,COST,LDI)[LJ]   GPO         4013     A
GPO                LX7    I.COSTP                                             GPO         4014     A
GPO                IX6    X7+X6       COST[LI] = COST[LI] + 1                 GPO         4015     A
GPO                SA6    A4                                                  GPO         4016     A
GPO                EQ     MIP2                                                GPO         4017          I
          -FCC0717
GPO                EQ     MIP1                                                FCC0717       59     A
GPO        IM      SPACE  3,10                                                GPO         4018     A
GPO                PROCESS IM                                                 GPO         4019     A
GPO                LX5    -R1.RJP                                             GPO         4020     A
GPO                BX6    "RN"X5                                              GPO         4021     A
GPO                LX5    R1.RJP-R1.RKP                                       GPO         4022     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                SA2    "TB"+X6     LJ = TB + RJ[R1]                          GPO          4023     A
GPO                BX7    "RN"X5                                                GPO          4024     A
GPO                SA3    "TB"+X7     LK = TB + RK[R1]                          GPO          4025     A
GPO                BX6    X2*X3                                                 GPO          4026     A
GPO                LX6    59-I.IPP                                              GPO          4027     A
GPO                PL     X6,MIP2     IF ^( IP[LJ] & IP[LK] )                   GPO          4028     A
GPO                                                                             GPO          4029     A
GPO                BX7    X2+X3       LJK = [LJ] ! [LK]                         GPO          4030     A
GPO                SX5    B3          ICOST = 3                                 GPO          4031     A
GPO                LX7    59-I.INVP                                             GPO          4032     A
GPO                MI     X7,IA1      IF INV[LJ] ! INV[LK]                      GPO          4033          I
          -FCC2271X
GPO                                                                             GPO          4034          I
          -FCC2271X
GPO                EQ     MIP2                                                  GPO          4035          I
          -FCC2271X
GPO                PL     X7,MIP2     IF NEITHER INVARIANT                      FCC2271X        8     A
GPO                SA4    AN                                                    FCC2271X        9     A
GPO                MI     X4,IM5      IF UNCONDTIONALLY EXECUTED                FCC2271X       10     A
GPO                SA2    A2-B1       DJ = LJ - 1                               FCC2271X       11     A
GPO                CALL   CMO         CHECK SAFETY OF J OPERAND                 FCC2271X       12     A
GPO                SA2    A3-B1       DK = LK - 1                               FCC2271X       13     A
GPO                ZR     X6,IM1      IF J IS SAFE                              FCC2271X       14     A
GPO                CALL   CMO         CHECK SAFETY OF K OPERAND                 FCC2271X       15     A
GPO                ZR     X6,IM2      IF K IS SAFE                              FCC2271X       16     A
GPO                EQ     MIP2        NOT SAFE (NEITHER OPERAND)                FCC2271X       17     A
GPO                                                                             FCC2271X       18     A
GPO       IM1      CALL   CMO         CHECK SAFETY OF K (J IS SAFE)             FCC2271X       19     A
GPO                ZR     X6,IM4      IF K IS SAFE                              FCC2271X       20     A
GPO                                                                             FCC2271X       21     A
GPO       *        ONE OPERAND SAFE.  MARK MULTIPLY UNSAFE SO *SQZ* WILL       FCC2271X       22     A
GPO       *        INSERT AN UNPACK, AND HOIST.                                FCC2271X       23     A
GPO                                                                             FCC2271X       24     A
GPO       IM2      SA4    A5+2        DI = RI + 2                               FCC2271X       25     A
GPO                SX6    B1                                                    FCC2271X       26     A
GPO                LX6    D.USIP                                                FCC2271X       27     A
GPO                BX6    X6+X4       USI[DI] = 1                               FCC2271X       28     A
GPO                SA6    A4                                                    FCC2271X       29     A
GPO                                                                             FCC2271X       30     A
GPO       *        MULTIPLY CAN BE HOISTED WITH IMPUNITY.                      FCC2271X       31     A
GPO                                                                             FCC2271X       32     A
GPO       IM4      SA5    A5                                                    FCC2271X       33     A
GPO                LX5    -R1.RJP                                               FCC2271X       34     A
GPO                BX6    "RN"X5                                                FCC2271X       35     A
GPO                SA2    "TB"+X6     RESTORE LJ                                FCC2271X       36     A
GPO                SA4    A5+3        RETORE LI                                 FCC2271X       37     A
GPO                BX7    X2+X3       RESTORE LJK                               FCC2271X       38     A
GPO                SX5    B3          ICOST = 3                                 FCC2271X       39     A
GPO                EQ     IA1                                                   FCC2271X       40     A
GPO                                                                             FCC2271X       41     A
GPO       IM5      SA4    A5+3        RESTORE LI                                FCC2271X       42     A
GPO                EQ     IA1                                                   FCC2271X       43     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO        ADDS     EJECT                                                    GPO        4036    A
GPO       **        ADDS - ONLY ALLOW *IP*"S IN 1 VARIABLE                   GPO        4037    A
GPO                                                                          GPO        4038    A
GPO                 PROCESS (IA,IS)                                          GPO        4039    A
GPO                 SA3    A4-B1       DI = LI - 1                           GPO        4040    A
GPO                 LX5    -R1.RJP                                           GPO        4041    A
GPO                 BX6    "RN"X5                                            GPO        4042    A
GPO                 LX5    R1.RJP-R1.RKP                                     GPO        4043    A
GPO                 BX7    "RN"X5                                            GPO        4044    A
GPO                 SA2    "TB"+X6     LJ = TB + RJ[R1]                      GPO        4045    A
GPO                 LX5    B4,X3                                             GPO        4046    A
GPO                 SA3    "TB"+X7     LK = TB + RK[R1]                      GPO        4047    A
GPO                 BX6    X2*X3                                             GPO        4048    A
GPO                 LX6    59-I.IPP                                          GPO        4049    A
GPO                 PL     X6,IA3      IF ^( IP[LJ] & IP[LK] )               GPO        4050    A
GPO                                                                          GPO        4051    A
GPO                 BX7    X2+X3       LJK = [LJ] ! [LK]                     GPO        4052    A
GPO                 MI     X5,IA2      IF INC[DI]                            GPO        4053    A
GPO                 SX5    B1          ICOST = 1                             GPO        4054    A
GPO                 LX7    59-I.INVP                                         GPO        4055    A
GPO                 MI     X7,IA1      IF INV[LJ] ! INV[LK]                  GPO        4056    A
GPO                                                                          GPO        4057    A
GPO       *         BOTH ^INV, MAKE SURE VARIABLES ARE THE SAME              GPO        4058    A
GPO                                                                          GPO        4059    A
GPO                 BX6    X2-X3                                             GPO        4060    A
GPO                 LX6    -I.LDIP                                           GPO        4061    A
GPO                 SB2    X6                                                GPO        4062    A
GPO                 NZ     B2,MIP2     IF LDI[LJ] " LDI[LK]                  GPO        4063    A
GPO                                                                          GPO        4064    A
GPO                 NZ     X6,IA1      IF [LJ] " [LK]                        GPO        4065    A
GPO                 MX3    0           COST[LK] = 0  */ IN REGS ONLY         GPO        4066    A
GPO                                                                          GPO        4067    A
GPO       *         ADD INSTRUCTION TO *IP* LIST, COST = ICOST + SIGMA(COST[PRED])  GPO  4068    A
GPO                                                                          GPO        4069    A
GPO        IA1      BX6    -X1*X2                                            GPO        4070    A
GPO                 SA2    MIPD                                              GPO        4071    A
GPO                 LX5    I.COSTP                                           GPO        4072    A
GPO                 BX3    -X1*X3                                            GPO        4073    A
GPO                 IX4    X4+X5       COST[LI] = ICOST                      GPO        4074    A
GPO                 LX7    1+I.INVP                                          GPO        4075    A
GPO                 BX7    X2*X7       IP[LI] = 1;  LDI[LI] = LDI[LJK]       GPO        4076    A
GPO                 IX6    X3+X6                                             GPO        4077    A
GPO                 BX4    X7+X4                                             GPO        4078    A
GPO                 IX6    X6+X4       COST[LI] = COST[LI] + COST[LJ] + COST[LK]  GPO   4079    A
GPO                 SA6    A4+                                               GPO        4080    A
GPO                 EQ     MIP1                                              GPO        4081    A
GPO                                                                          GPO        4082    A
GPO       *         INCREMENT, SET COST = 1, SINCE ANY USES OF IT ARE USES OF A  GPO   4083    A
GPO       *         *LD* OF THE REDEFINED VARIABLE.                          GPO        4084          I
          -FCC0656
GPO       *         LD OF THE REDEF VAR, BUMP LDI TO INDICATE NEW VARIABLE.  FCC0656      44    A
GPO                                                                          GPO        4085    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO       IA2      SA2    MIPC                                                  GPO         4086     A
GPO                LX7    -I.LDIP                                               GPO         4087     A
GPO                SX7    X7                                                    GPO         4088          I
          -FCC0656
GPO                SX7    X7+B1       LDI[LI] = LDI[LJK] + 1 */ MARK NEW VAR    FCC0656       45     A
GPO                LX7    I.LDIP                                                GPO         4089     A
GPO                BX4    X7+X4       LDI[LI] = LDI[LJK]                        GPO         4090          I
          -FCC0656
GPO                BX4    X7+X4                                                 FCC0656       46     A
GPO                BX6    X2+X4       IP[LI] = 1;  COST[LI] = 1                 GPO         4091     A
GPO                SA6    A4                                                    GPO         4092     A
GPO                EQ     MIP1                                                  GPO         4093     A
GPO       IA       SPACE  1,10                                                  GPO         4094     A
GPO      *         CHECK FOR *IA* + *INV*, AND MARK *IA* IF SO                  GPO         4095     A
GPO                                                                             GPO         4096     A
GPO       IA3      BX7    X2+X3                                                 GPO         4097     A
GPO                LX7    59-I.INVP                                             GPO         4098     A
GPO                PL     X7,MIP2     IF ^( INV[LJ] ! INV[LK] )                 GPO         4099     A
GPO                LX7    I.INVP-I.IAP                                          GPO         4100     A
GPO                PL     X7,MIP2     IF ^( IA[LJ] ! IA[LK] )                   GPO         4101     A
GPO                                                                             GPO         4102     A
GPO                SX7    B1                                                    GPO         4103     A
GPO                LX7    I.IAP                                                 GPO         4104     A
GPO                BX6    X7+X4       IA[LI] = 1                                GPO         4105     A
GPO                SA6    A4                                                    GPO         4106     A
GPO                EQ     MIP2                                                  GPO         4107     A
GPO       STT      SPACE  2,14                                                  GPO         4108     A
GPO      *         STT - PROPAGATE *IA* BIT UP FOR USE BY *JPX* PROCESSOR       GPO         4109     A
GPO                                                                             GPO         4110     A
GPO                PROCESS STT                                                  GPO         4111     A
GPO                SA3    A5+B1       R2 = R1 + 1                               GPO         4112     A
GPO                SX7    B1                                                    GPO         4113     A
GPO                LX3    -IH.IHP                                               GPO         4114     A
GPO                SB2    X3                                                    GPO         4115     A
GPO                LX3    IH.IHP-IH.RFP                                         GPO         4116     A
GPO                LX7    I.IAP                                                 GPO         4117     A
GPO                BX6    "RN"X3                                                GPO         4118     A
GPO                SA2    "TB"+X6     LIP = TB + RF[R2]                         GPO         4119     A
GPO                BX7    X7*X2                                                 GPO         4120     A
GPO                BX6    X7+X4       IA[LI] = IA[LIP]                          GPO         4121     A
GPO                SA6    A4                                                    GPO         4122     A
GPO                                                                             GPO         4123     A
GPO      *         ADD TO *IP* LIST IF NOT *PS* OR *PRS* ( UNPROFITABLE CASES )  GPO         4124     A
GPO                                                                             GPO         4125     A
GPO                SA3    A3+B1       DI = R2 + 1                               GPO         4126     A
GPO                MX7    -D.STRSL                                              GPO         4127     A
GPO                LX3    -D.STRSP                                              GPO         4128          I
          -FCC0717
GPO                LX7    D.STRSP                                               FCC0717       60     A
GPO                BX7    -X7*X3                                                GPO         4129     A
GPO                NZ     X7,MIP2     IF STRS[DI] " 0  */ *INC* OR INIT CODE    GPO         4130     A
GPO                ZR     B2,STT1     IF IH[R2] = 0                             GPO         4131     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                                                                                GPO        4132     A
GPO                 SB2    A2-"TB"                                                 GPO        4133     A
GPO                 NZ     B2,MIP2     IF RF[R2] " 0                               GPO        4134     A
GPO                 EQ     .S                                                      GPO        4135     A
GPO                                                                                GPO        4136     A
GPO        STT1     LX2    59-I.IPP                                                GPO        4137     A
GPO                 PL     X2,MIP2     IF ^IP[LIP]                                 GPO        4138     A
GPO                 LX2    1+I.IPP                                                 GPO        4139     A
GPO                 SA3    MIPA                                                    GPO        4140     A
GPO                 BX7    X2*X3                                                   GPO        4141     A
GPO                 BX6    X6+X7       (IP,COST,LDI)[LI] = (IP,COST,LDI)[LIP]      GPO        4142     A
GPO                 SA6    A6                                                      GPO        4143     A
GPO                 EQ     MIP1                                                    GPO        4144     A
GPO        JPX      EJECT                                                          GPO        4145     A
GPO       *         JPX - CHECK FOR LOOP BACK JUMP, AND IF SO COLLECT INFORMATION  GPO        4146     A
GPO       *         FOR TEST REPLACEMENT IN *GRA*.  STATEMENTS OF THE FORM -       GPO        4147     A
GPO       *         IF( I @ LIM ) GO TO LIH  RI = LIM - I ;  PL  RI,LIH            GPO        4148     A
GPO       *         IF( I < LIM ) GO TO LIH  RI = I - LIM ;  MI  RI,LIH            GPO        4149     A
GPO       *         ARE THE PATTERNS THAT WE LOOK FOR HERE.                        GPO        4150     A
GPO                                                                                GPO        4151     A
GPO                 PROCESS JPX                                                    GPO        4152     A
GPO                 SA2    N.LJ                                                    GPO        4153     A
GPO                 SA3    LIH                                                     GPO        4154     A
GPO                 SA4    A5+B1       R2 = R1 + 1                                 GPO        4155     A
GPO                 SB2    X2                                                      GPO        4156     A
GPO                 BX7    X3-X4                                                   GPO        4157     A
GPO                 SX6    X7                                                      GPO        4158     A
GPO                 NE     B2,B1,MIP2  IF N.LJ " 1                                 GPO        4159     A
GPO                 NZ     X6,MIP2     IF IH[R2] " LIH                             GPO        4160     A
GPO                 LX4    -IH.CAP                                                 GPO        4161     A
GPO                 SB4    X4-2        JPT = CA[R2] - 2                            GPO        4162     A
GPO                 SB2    "TB"-B3                                                 GPO        4163     A
GPO                 MI     B4,MIP2     IF JPT @ 0    */ ZR, NZ JUMPS               GPO        4164     A
GPO                 GT     B4,B1,MIP2  IF JPT > 1    */ RANGE CHECK JUMPS          GPO        4165     A
GPO                                                                                GPO        4166     A
GPO       *         FIND UPPER LIMIT VALUE AND SAVE IN *LUL*                       GPO        4167     A
GPO                                                                                GPO        4168     A
GPO                 SA4    B2+X5       RIP = TB-3 + RI[R1]                         GPO        4169     A
GPO                 UX6    B2,X4                                                   GPO        4170     A
GPO                 SX7    B2-OC.STT                                               GPO        4171     A
GPO                 NZ     X7,JPX1     IF OC[R1P] " OC.STT                         GPO        4172     A
GPO                                                                                GPO        4173     A
GPO                 NE     B4,B1,MIP2  IF JPT " 1    */ NOT *MI* JUMP              GPO        4174     A
GPO                 SA3    A4+B1       R2P = R1P + 1                               GPO        4175     A
GPO                 LX3    -IH.CAP                                                 GPO        4176     A
GPO                 MX2    IH.CAL                                                  GPO        4177     A
GPO                 SX4    X3+B1                                                   GPO        4178     A
GPO                 LX2    IH.CAL                                                  GPO        4179     A
GPO                 BX7    -X4*X2                                                  GPO        4180     A
GPO                 PX6    X7          ULW = PACK( 0 , -(CA[R2P]+1) )              GPO        4181     A
GPO                 LX3    IH.CAP-IH.RFP                                           GPO        4182     A
GPO                 BX5    "RN"X3      R = RF[R2P]                                 GPO        4183     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                EQ     JPX5                                                  GPO        4184     A
GPO                                                                             GPO        4185     A
GPO        JPX1    SX7    B2-OC.IS                                              GPO        4186     A
GPO                NZ     X7,MIP2      IF OC[R1P] " OC.IS                       GPO        4187     A
GPO                                                                             GPO        4188     A
GPO        *       CHECK LIMIT OPERAND OF *IS*                                  GPO        4189     A
GPO                                                                             GPO        4190     A
GPO                SA3    A4+B3        LIP = R1P + 3                            GPO        4191     A
GPO                MX7    2                                                     GPO        4192     A
GPO                LX3    58-I.IAIPP                                            GPO        4193     A
GPO                BX7    X3*X7                                                 GPO        4194     A
GPO                ZR     X7,MIP2      IF ^( IA[LIP] ! IP[LIP] )                GPO        4195     A
GPO                                                                             GPO        4196     A
GPO                LX4    -R1.RJP                                               GPO        4197     A
GPO                NZ     B4,JPX2      IF JPT " 0                               GPO        4198     A
GPO                                                                             GPO        4199     A
GPO                BX7    "RN"X4       LRN = RJ[R1P] */ LIMIT R-NUMBER          GPO        4200     A
GPO                LX4    R1.RJP-R1.RKP                                         GPO        4201     A
GPO                BX5    "RN"X4       R = RK[R1P]                              GPO        4202     A
GPO                EQ     JPX3                                                  GPO        4203     A
GPO                                                                             GPO        4204     A
GPO        JPX2    BX5    "RN"X4       R = RJ[R1P]                              GPO        4205     A
GPO                LX5    R1.RJP-R1.RKP                                         GPO        4206     A
GPO                BX7    "RN"X4       LRN = RK[R1P]                            GPO        4207     A
GPO                                                                             GPO        4208     A
GPO        JPX3    SB2    "TB"-B1                                               GPO        4209     A
GPO                SA2    B2+X7        DIP = TB-1 + LRN                         GPO        4210     A
GPO                SA4    A2-2         R1P = DIP - 2                            GPO        4211     A
GPO                LX2    59-D.ZPP                                              GPO        4212     A
GPO                PL     X2,MIP2      IF ^ZP[DIP]   */ NOT *LD* OR *S*         GPO        4213     A
GPO                LX2    D.ZPP-D.LDP                                           GPO        4214     A
GPO                PL     X2,JPX4      IF ^LD[DIP]                              GPO        4215     A
GPO                LX2    D.LDP-D.TYP-1                                         GPO        4216     A
GPO                PL     X2,MIP2      IF TYPE[DIP] " III                       GPO        4217     A
GPO                                                                             GPO        4218     A
GPO                NZ     B4,MIP2      IF JPT " 0    */ CANT ADD 1 TO UPPER LIMIT GPO     4219     A
GPO                SA3    A4+B1        R2P = R1P + 1                            GPO        4220     A
GPO                MX2    -IH.CAIHL                                             GPO        4221     A
GPO                LX4    -R1.INP                                               GPO        4222     A
GPO                SX4    X4                                                    GPO        4223     A
GPO                BX2    -X2*X3                                                GPO        4224     A
GPO                LX4    PS.UDIP                                               GPO        4225     A
GPO                BX7    X4+X2                                                 GPO        4226     A
GPO                PX6    B1,X7        ULW = PACK( 1 , IN[R1P] , CAIH[R2P] )    GPO        4227     A
GPO                EQ     JPX5                                                  GPO        4228     A
GPO                                                                             GPO        4229     A
GPO        JPX4    UX6    B2,X4                                                 GPO        4230     A
GPO                SX7    B2-OC.S                                               GPO        4231     A
GPO                LX4    -R1.INP                                               GPO        4232     A
GPO                NZ     X7,MIP2      IF OC[R1P] " OC.S                        GPO        4233     A
GPO                SB2    -B4                                                   GPO        4234     A
GPO                SX4    X4+B2                                                 GPO        4235     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                MX3    -IH.CAL                                                GPO        4236    A
GPO                BX7    -X3*X4                                                 GPO        4237    A
GPO                PX6    X7          ULW = PACK( 0 , IN[R1P]-JPT )              GPO        4238    A
GPO                                                                              GPO        4239    A
GPO      *         NOW CHECK FOR THE INDEX, IT MUST BE *RD* IN THE LOOP.          GPO        4240    A
GPO                                                                              GPO        4241    A
GPO       JPX5     SA4    "TB"+X5     LIP = TB + R                               GPO        4242    A
GPO                LX4    59-I.IAP                                               GPO        4243    A
GPO                SA6    LUL         LUL = ULW                                  GPO        4244    A
GPO                PL     X4,JPX7     IF ^IA[LIP]                                GPO        4245    A
GPO                                                                              GPO        4246    A
GPO      *         PRED IS *IA* ( IA,IS,STT ) , GET ^INV PRED OF THIS INSTRUCTION GPO        4247    A
GPO                                                                              GPO        4248    A
GPO                SA3    A4-B1       DIP = LIP - 1                              GPO        4249    A
GPO                LX3    58-D.TYP                                               GPO        4250    A
GPO                PL     X3,JPX6     IF TYPE[DIP] = I                           GPO        4251    A
GPO                SA2    A3-B1       R2P = DIP - 1                              GPO        4252    A
GPO                LX2    -IH.RFP                                                GPO        4253    A
GPO                BX5    "RN"X2      R = RF[R2P]                                GPO        4254    A
GPO                EQ     JPX7                                                   GPO        4255    A
GPO                                                                              GPO        4256    A
GPO       JPX6     SA3    A4-B3       R1P = LIP - 3                              GPO        4257    A
GPO                LX3    -R1.RJP                                                GPO        4258    A
GPO                BX5    "RN"X3      R = RJ[R1P]                                GPO        4259    A
GPO                SA4    "TB"+X5     LIP = TB + R                               GPO        4260    A
GPO                LX4    59-I.INVP                                              GPO        4261    A
GPO                PL     X4,JPX7     IF ^INV[LIP]                               GPO        4262    A
GPO                LX3    R1.RJP-R1.RKP                                          GPO        4263    A
GPO                BX5    "RN"X3      R = RK[R1P]                                GPO        4264    A
GPO                                                                              GPO        4265    A
GPO      *         CHECK FOR *LD* OF A *RD* VARIABLE                              GPO        4266    A
GPO                                                                              GPO        4267    A
GPO       JPX7     SB2    "TB"-B3                                                GPO        4268    A
GPO                SA4    B2+X5       R1P = TB-3 + R                             GPO        4269    A
GPO                UX6    B2,X4                                                  GPO        4270    A
GPO                LX4    -R1.INP                                                GPO        4271    A
GPO                SX7    B2-OC.LD                                               GPO        4272    A
GPO                NZ     X7,MIP2     IF OC[R1P] " OC.LD                         GPO        4273    A
GPO                SA2    "UB"+X4     UI = UB + IN[R1]                           GPO        4274    A
GPO                LX2    59-UD.^RDP                                             GPO        4275    A
GPO                MI     X2,MIP2     IF ^RD[UI]    */ VAR NOT ^RD IN LOOP       GPO        4276    A
GPO                                                                              GPO        4277    A
GPO      *         TEST REPLACEMENT POSSIBLE, SET FLAGS FOR *GRA*                 GPO        4278    A
GPO                                                                              GPO        4279    A
GPO                SA3    A4+B1       R2P = R1P + 1                              GPO        4280    A
GPO                MX2    -IH.CAIHL                                              GPO        4281    A
GPO                SA5    A5          R1 = R1                                    GPO        4282    A
GPO                SX4    X4                                                     GPO        4283    A
GPO                BX2    -X2*X3                                                 GPO        4284    A
GPO                SB2    "TB"-B1                                                GPO        4285    A
GPO                LX4    PS.UDIP                                                GPO        4286    A
GPO                BX7    X4+X2                                                  GPO        4287    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                SA4    B2+X5       DIP = TB-1 + RI[R1]                          GPO        4288     A
GPO                PX6    B1,X7                                                    GPO        4289     A
GPO                SA6    LCV         LCV = PACK( 1 , IN[R1P] , CAIH[R2P] )        GPO        4290     A
GPO                SX6    B1                                                       GPO        4291     A
GPO                SA6    LTT         LTT = 1       */ LOOP TEST TYPE              GPO        4292     A
GPO                LX6    D.INCP                                                   GPO        4293     A
GPO                BX6    X4+X6       INC[DIP] = 1  */ PRED OF *JPX* IS TEST USE   GPO        4294     A
GPO                SA6    A4                                                       GPO        4295     A
GPO                SA4    A5+2        DI = R1 + 2                                  GPO        4296     A
GPO                SX6    B1                                                       GPO        4297     A
GPO                LX6    D.INCP                                                   GPO        4298     A
GPO                BX6    X6+X4       INC[DI] = 1   */ MARK *JPX* AS LOOP TEST JUMP GPO        4299     A
GPO                SA6    A4                                                       GPO        4300     A
GPO                EQ     MIP2                                                     GPO        4301     A
GPO                                                                               GPO        4302     A
GPO                QUAL   *                                                        GPO        4303     A
GPO                                                                               GPO        4304     A
GPO       MIP      EQU    /MIP/MIP                                                GPO        4305     A
GPO       MIE      TITLE  MIE - MOVE INVARIANT EXPRESSIONS                        GPO        4306     A
GPO      **        MIE - MOVE INVARIANT EXPRESSIONS AND TERMINAL POLYNOMIALS TO   GPO        4307     A
GPO      *         HOLDING BLOCK.                                                 GPO        4308     A
GPO      *                                                                        GPO        4309     A
GPO      *         THE CHIEF CONCERN OF *MIE* IS PROFIT. THE CRITERIA USED HERE   GPO        4310     A
GPO      *         IS TO MOVE OUT AN INSTRUCTION IF IT HAS PREDECESSORS AND IS    GPO        4311     A
GPO      *         NOT AN *XMT* THAT *PS* OR A *SA*/*STT* THAT PRECEEDS A TEMP    GPO        4312     A
GPO      *         LOCK *RS* TO A B-REG.                                          GPO        4313     A
GPO      *                                                                        GPO        4314     A
GPO      *         SINCE THE MOTION OF *IP*"S INVOLVES PLACING AN INCREMENT       GPO        4315     A
GPO      *         INSTRUCTION BACK IN THE LOOP, WE USE THE *COST* OF THE *IP*    GPO        4316     A
GPO      *         AS THE CHIEF PROFIT CRITERIA.  THE CONDITION THAT A *IP* BE    GPO        4317     A
GPO      *         *RF* IS NECESSARY FOR THE *IP* TO BE A B-REGISTER CANIDATE.    GPO        4318     A
GPO      *                                                                        GPO        4319     A
GPO      *         EXIT   INVARIANT EXPRESSIONS, TERMINAL POLYNOMIALS CODE MOVED  GPO        4320     A
GPO      *         TO HOLDING BLOCK ( *TXT* ). *TET* UPDATED.                     GPO        4321     A
GPO                                                                               GPO        4322     A
GPO       MIE      ROUTINE                                                        GPO        4323     A
GPO                SA3    MINC                                                    GPO        4324     A
GPO                S"TB"  X5+3        TB = TBA + 3                                GPO        4325     A
GPO                LX4    -BS.IICP                                                GPO        4326     A
GPO                SA0    B0          LII = 0                                     GPO        4327     A
GPO                MX1    -D.USESL                                                GPO        4328     A
GPO                SB2    X3                        (B2) = MINC                   GPO        4329     A
GPO                SB4    B0          NTI = 0       */ N.TERMINAL INST            GPO        4330     A
GPO                SB5    B0          NMI = 0       */ N.MOVED INST               GPO        4331     A
GPO                SB6    X4          I = IIC[BSW]  */ NEXT ON *IIC* CHAIN        GPO        4332     A
GPO                                                                               GPO        4333     A
GPO      *         RESCAN *IIC* AND FORM A FOWARD CHAIN OF INST THAT ARE PROFITAB GPO        4334     A
GPO      *         TO MOVE, OR USED BY ANOTHER INSTRUCTION.                       GPO        4335     A
GPO                                                                               GPO        4336     A
GPO       MIE1     SA5    "TB"+B6     LI = TB + I                                 GPO        4337     A
GPO                ZR     B6,MIE14    IF I = 0      */ END OF CHAIN               GPO        4338     A
GPO                SA4    A5-B1       DI = LI - 1                                 GPO        4339     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                BX3    X5                                                  GPO        4340    A
GPO                LX5    -I.USESP                                            GPO        4341    A
GPO                BX7    -X1*X5                                              GPO        4342    A
GPO                LX5    I.USESP-I.IICP                                      GPO        4343    A
GPO                BX6    X4          D = [DI]                                GPO        4344    A
GPO                SB6    X5          I = IIC[LI]   */ NEXT ON CHAIN          GPO        4345    A
GPO                NZ     X7,MIE5     IF USES[LI] " 0  */ USED BY ANOTHER     GPO        4346    A
GPO                                                                           GPO        4347    A
GPO                LX3    59-I.INVP                                           GPO        4348    A
GPO                MI     X3,MIE2     IF INV[LI]                              GPO        4349    A
GPO                                                                           GPO        4350    A
GPO       *        TERMINAL *IP*, SEE IF PROFITABLE TO MOVE                    GPO        4351    A
GPO                                                                           GPO        4352    A
GPO                LX6    59-D.RFP                                            GPO        4353    A
GPO                LX3    1+I.INVP-I.COSTP                                    GPO        4354    A
GPO                PL     X6,MIE1     IF ^RF[D]                               GPO        4355         I
          -CCG21
GPO                BX7    -X1*X3                                              GPO        4356    A
GPO                SB3    X7                                                  GPO        4357    A
GPO                GE     B3,B2,MIE7  IF COST[LI] \ MINC                      GPO        4358         I
          -CCG21
GPO                LT     B3,B2,MIE1  IF COST[LI] < MINC                      CCG21        18    A
GPO                MI     X6,MIE7     IF RF[D]                                CCG21        19    A
GPO                SB3    B3-4                                                CCG21        20    A
GPO                PL     B3,MIE7     IF COST[LI] \ 4                         CCG21        21    A
GPO                                                                           GPO        4359    A
GPO                EQ     MIE1                                                GPO        4360    A
GPO        MIE     SPACE  2,14                                                GPO        4361    A
GPO       *        TERMINAL INV INST, SEE IF PROFITABLE TO MOVE                GPO        4362    A
GPO                                                                           GPO        4363    A
GPO        MIE2    LX6    58-D.STRSP                                          GPO        4364    A
GPO                MX7    D.STRSL                                             GPO        4365    A
GPO                BX3    X7*X6                                               GPO        4366    A
GPO                NZ     X3,MIE3     IF STRS[D] " 0                          GPO        4367    A
GPO                LX6    1+D.STRSP-D.ZPP                                     GPO        4368    A
GPO                MI     X6,MIE1     IF ZP[D]      */ NO PRED                GPO        4369    A
GPO                EQ     MIE7                                                GPO        4370    A
GPO                                                                           GPO        4371    A
GPO        MIE3    LX6    1+D.STRSP-D.PSP                                     GPO        4372    A
GPO                MI     X6,MIE4     IF PS[DI]                               GPO        4373    A
GPO                SA3    A5+4                                                GPO        4374    A
GPO                SX7    B1          IU = 1                                  GPO        4375    A
GPO                LX3    59-I.INVP                                           GPO        4376    A
GPO                MI     X3,MIE6     IF INV[LI+4]  */ INV FUNCTION ARG       GPO        4377    A
GPO                SA2    A5+B1       R1RS = LI + 1                           GPO        4378    A
GPO                LX2    59-R1.SOP-4                                         GPO        4379    A
GPO                PL     X2,MIE1     IF SORT[R1RS] " 2  */ ^ TO AN X-REGISTER GPO       4380    A
GPO                                                                           GPO        4381    A
GPO       *        ( PS ! PRS ) & USES = 0                                     GPO        4382    A
GPO                                                                           GPO        4383    A
GPO        MIE4    LX6    D.PSP-D.ZPP                                         GPO        4384    A
GPO                MI     X6,MIE1     IF ZP[D]                                GPO        4385    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO              LX6    D.ZPP-D.XMTP                                          GPO        4386    A
GPO              MI     X6,MIE1     IF XMT[D]                                 GPO        4387    A
GPO              EQ     MIE7                                                  GPO        4388    A
GPO       MIE    SPACE  2,10                                                  GPO        4389    A
GPO      *       INSTRUCTION IS OPERAND OF ANOTHER THAT HAS BEEN MOVED OUT.   GPO        4390    A
GPO                                                                           GPO        4391    A
GPO       MIE5   LX6    59-D.PRSP                                             GPO        4392    A
GPO              PL     X6,MIE7     IF ^PRS[D]                                GPO        4393    A
GPO                                                                           GPO        4394    A
GPO              SA2    A5+B1       R1RS = LI + 1                             GPO        4395    A
GPO              SA3    A5+4                                                  GPO        4396    A
GPO              MX7    0           IU = 0                                    GPO        4397    A
GPO              LX3    59-I.INVP                                             GPO        4398    A
GPO              LX2    59-R1.SOP-SO.LKP                                      GPO        4399    A
GPO              MI     X2,MIE6     IF SOLK[R1RS] " 0  */ TEMP LOCK           GPO        4400    A
GPO              PL     X3,MIE7     IF ^INV[LI+4]                             GPO        4401    A
GPO              SX7    B1          IU = 1                                    GPO        4402    A
GPO                                                                           GPO        4403    A
GPO      *       ADD *RS* TO CHAIN OF MOVED INST                              GPO        4404    A
GPO                                                                           GPO        4405    A
GPO       MIE6   LX3    1+I.INVP-I.IICP                                       GPO        4406    A
GPO              SX6    A0                                                    GPO        4407    A
GPO              BX3    X0*X3                                                 GPO        4408    A
GPO              IX6    X6+X3       IIC[LI+4] = LII                           GPO        4409    A
GPO              LX6    I.IICP                                                GPO        4410    A
GPO              SB5    B5+B1       NMI = NMI + 1                             GPO        4411    A
GPO              SA6    A3                                                    GPO        4412    A
GPO              SA0    A3-"TB"     LII = LI+4 - TB                           GPO        4413    A
GPO              LX7    I.USESP-I.IICP                                        GPO        4414    A
GPO              IX5    X7+X5       USES[LI] = USES[LI] + IU                  GPO        4415    A
GPO       MIE    SPACE  1,10                                                  GPO        4416    A
GPO      *       ADD INSTRUCTION TO CHAIN OF MOVED INSTRUCTIONS, MARK AS      GPO        4417    A
GPO      *       TERMINAL IF NOT ALL USES ARE CONTAINED IN THE *INV* CHAIN.   GPO        4418    A
GPO                                                                           GPO        4419    A
GPO       MIE7   SX3    A0                                                    GPO        4420    A
GPO              BX5    X0*X5                                                 GPO        4421    A
GPO              IX6    X3+X5                                                 GPO        4422    A
GPO              SB5    B5+B1       NMI = NMI + 1                             GPO        4423    A
GPO              LX6    I.IICP      IIC[LI] = LII                             GPO        4424    A
GPO              SA0    A5-"TB"     LII = LI - TB                             GPO        4425    A
GPO              SA6    A5                                                    GPO        4426    A
GPO              LX5    I.IICP-I.USESP+D.USESP                                GPO        4427    A
GPO              BX7    X5-X4                                                 GPO        4428    A
GPO              LX7    -D.USESP                                              GPO        4429    A
GPO              BX2    -X1*X7                                                GPO        4430    A
GPO              ZR     X2,MIE8     IF USES[LI] = USES[DI]                    GPO        4431    A
GPO                                                                           GPO        4432    A
GPO              SX7    B1                                                    GPO        4433    A
GPO              SB4    B4+B1       NTI = NTI + 1                             GPO        4434    A
GPO              LX7    I.TIP                                                 GPO        4435    A
GPO              BX6    X7+X6       TI[LI] = 1    */ MARK AS TERMINAL         GPO        4436    A
GPO              SA6    A6                                                    GPO        4437    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO       MIE      SPACE  1,10                                                GPO        4438     A
GPO      *         BUMP USES COUNT OF OPERANDS OF INSTRUCTION                 GPO        4439     A
GPO                                                                           GPO        4440     A
GPO       MIE8     LX4    58-D.TYP                                            GPO        4441     A
GPO                IX7    X4+X4                                               GPO        4442     A
GPO                SX6    B1                                                  GPO        4443     A
GPO                MI     X7,MIE1     IF TYPE[DI] = II ! IV                   GPO        4444     A
GPO                LX6    I.USESP                                             GPO        4445     A
GPO                PL     X4,MIE10    IF TYPE[DI] = I                         GPO        4446     A
GPO                                                                           GPO        4447     A
GPO      *         TYPE III INSTRUCTION                                       GPO        4448     A
GPO                                                                           GPO        4449     A
GPO                SA3    A4-B1       R2 = DI - 1                             GPO        4450     A
GPO                LX4    1+D.TYP-D.^DP                                       GPO        4451     A
GPO                LX3    -IH.RFP                                             GPO        4452     A
GPO                BX7    "RN"X3      R = RF[R2]                              GPO        4453     A
GPO                SA2    "TB"+X7     L = TB + R                              GPO        4454     A
GPO                ZR     X7,MIE9     IF R @ 1                                GPO        4455     A
GPO                IX7    X6+X2       USES[L] = USES[L] + 1                   GPO        4456     A
GPO                SA7    A2                                                  GPO        4457     A
GPO                                                                           GPO        4458     A
GPO       MIE9     PL     X4,MIE1     IF DEF[DI]    */ NOT A STORE            GPO        4459     A
GPO                SA3    A3-B1       R1 = R2 - 1                             GPO        4460     A
GPO                BX7    "RN"X3                                              GPO        4461     A
GPO                SA2    "TB"+X7     L = TB + RI[R1]                         GPO        4462     A
GPO                IX7    X6+X2       USES[L] = USES[L] + 1                   GPO        4463     A
GPO                SA7    A2+                                                 GPO        4464     A
GPO                EQ     MIE1                                                GPO        4465     A
GPO                                                                           GPO        4466     A
GPO      *         TYPE I INSTRUCTION                                         GPO        4467     A
GPO                                                                           GPO        4468     A
GPO       MIE10    MX7    -D.TYIOL                                            GPO        4469     A
GPO                LX4    2+D.TYP-D.TYIOP                                     GPO        4470     A
GPO                BX2    -X7*X4                                              GPO        4471     A
GPO                SA3    =XTYIOM+X2                                          GPO        4472     A
GPO                SA5    A4-2        R1 = DI - 2                             GPO        4473     A
GPO                UX7    B3,X3                                               GPO        4474     A
GPO                BX2    X7*X5                                               GPO        4475     A
GPO                AX5    B3,X2       RNUMS = TYIOM[DI] & [R1]                GPO        4476     A
GPO                                                                           GPO        4477     A
GPO       MIE11    BX4    "RN"X5      R = RI[RNUMS]                           GPO        4478     A
GPO                AX5    R1.RIL      RNUMS = SHIFT( RNUMS , -16 )            GPO        4479     A
GPO                ZR     X4,MIE12    IF R = 0                                GPO        4480     A
GPO                SA3    "TB"+X4     L = TB + R                              GPO        4481     A
GPO                IX7    X6+X3       USES[L] = USES[L] + 1                   GPO        4482     A
GPO                SA7    A3+                                                 GPO        4483     A
GPO       MIE12    NZ     X5,MIE11    IF RNUMS " 0                            GPO        4484     A
GPO                                                                           GPO        4485     A
GPO                EQ     MIE1                                                GPO        4486     A
GPO       MIE      EJECT                                                      GPO        4487     A
GPO      *         MOVE THE FINAL CHAIN OF INSTRUCTIONS TO THE HOLDING BLOCK  GPO        4488     A
GPO                                                                           GPO        4489     A

                   0         1         2         3         4         5         6         7         8
                   12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO       MIE14    ZR      B5,MIE      IF NMI = 0    */ NOTHING MOVED                GPO          4490     A
GPO                SB4     B4+B4                                                     GPO          4491     A
GPO                SX2     B5+B4                                                     GPO          4492     A
GPO                LX2     2                                                         GPO          4493     A
GPO                SX7     A0                                                        GPO          4494     A
GPO                SA7     IC          IC = LII                                      GPO          4495     A
GPO                SX0     "TB"                                                      GPO          4496          I
          -FCC1845X
GPO                SX5     A0                                                        GPO          4497     A
GPO                ALLOC   TXT,X2+B1   ALLOC( TXT , 4*(NMI+2*NTI) + 1 )              GPO          4498     A
GPO                S"TB"   X0          HBI = OLD(L.TXT)                              GPO          4499          I
          -FCC1845X
GPO                SA4     O.SEQ       HBI = OLD(L.TXT)                              FCC1845X        5     A
GPO                S"TB"   X4+3                                                      FCC1845X        6     A
GPO                SA4     L.TET                                                     GPO          4500     A
GPO                SB3     X5          I = LII                                       GPO          4501     A
GPO                SA1     DMASK                                                     GPO          4502     A
GPO                SB4     X4          TO = L.TET    */ TEMP ORDINAL                 GPO          4503     A
GPO                SB5     X2          HB = O.TXT                                    GPO          4504     A
GPO                MX0     -R1.RIL+1                                                 GPO          4505          I
          -CCG11
GPO                SA0     B0          TSC = 0                                       GPO          4506     A
GPO                SX6     X2+B6                                                     GPO          4507     A
GPO                LX0     1                                                         GPO          4508          I
          -CCG11
GPO                SX0     RN.MASK                                                   CCG11         118     A
GPO                SA6     TBB         TBB = O.TXT + HBI                             GPO          4509     A
GPO                EQ      MIE16                                                     GPO          4510     A
GPO                                                                                  GPO          4511     A
GPO       *        INCREMENT MOVED, CHANGE IT TO AN *XMT* OF THE *RD* VARIABLE       GPO          4512     A
GPO       *        NOTE THAT AN *INC* CANNOT BE TERMINAL OR *PRS*                    GPO          4513     A
GPO                                                                                  GPO          4514     A
GPO       MIE14A   SA4     F.RDT+OC.XMT                                              GPO          4515     A
GPO                SB2     OC.XMT                                                    GPO          4516     A
GPO                BX7     X4                                                        GPO          4517     A
GPO                SA7     A6-B1       [HA-1] = RDT(OC.MXT)                          GPO          4518     A
GPO                LX3     -IH.RFP                                                   GPO          4519     A
GPO                SA6     A7-B1       [HA-2] = 0    */ R2 WORD                      GPO          4520     A
GPO                BX7     "RN"X2                                                    GPO          4521     A
GPO                SX6     X3          R = RF[R2]                                    GPO          4522     A
GPO                NZ      X6,MIE14B   IF R " 0                                      GPO          4523     A
GPO                LX2     -R1.RJP                                                   GPO          4524     A
GPO                BX6     "RN"X2      R = RJ[R1]                                    GPO          4525     A
GPO                SA3     "TB"+X6     LIP = TB + R                                  GPO          4526     A
GPO                LX3     59-I.INVP                                                 GPO          4527     A
GPO                PL      X3,MIE14B   IF ^INV[LIP]                                  GPO          4528     A
GPO                LX2     R1.RJP-R1.RKP                                             GPO          4529     A
GPO                BX6     "RN"X2      R = RK[R1]                                    GPO          4530     A
GPO                                                                                  GPO          4531     A
GPO       MIE14B   LX6     R1.RJP                                                    GPO          4532     A
GPO                PX7     B2,X7                                                     GPO          4533     A
GPO                BX6     X7+X6                                                     GPO          4534     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                SA6    A6-B1       [HA-3] = TYI(OC.XMT,R,0,RI[R1])                GPO        4535    A
GPO                                                                                  GPO        4536    A
GPO                                                                                  GPO        4537    A
GPO      *         ADVANCE TO NEXT ON CHAIN                                          GPO        4538    A
GPO                                                                                  GPO        4539    A
GPO       MIE15    LX5    -I.IICP                                                    GPO        4540    A
GPO                SB3    X5          I = IIC[LI]   */ *BLK* INDEX OF INSTRUCTION    GPO        4541    A
GPO                ZR     B3,MIE20    IF I = 0                                       GPO        4542    A
GPO                                                                                  GPO        4543    A
GPO      *         MOVE INSTRUCTION TO HOLDING BLOCK                                 GPO        4544    A
GPO                                                                                  GPO        4545         I
         -FCC1254
GPO       MIE16    SA5    "TB"+B3     LI = TB + I                                    GPO        4546         I
         -FCC1254
GPO                SA4    A5-B1       DI = LI - 1;  D = [DI]                         GPO        4547         I
         -FCC1254
GPO                SA3    A4-B1       R2 = DI - 1                                    GPO        4548         I
         -FCC1254
GPO                SA2    A3-B1       R1 = R2 - 1                                    GPO        4549         I
         -FCC1254
GPO                BX6    X2                                                         GPO        4550         I
         -FCC1254
GPO                SA6    B5+B6       HA = HB + HBI;  [HA] = [R1]                    GPO        4551         I
         -FCC1254
GPO                LX7    X3                                                         GPO        4552         I
         -FCC1254
GPO                SA7    A6+B1       HA = HA + 1;  [HA] = [R2]                      GPO        4553         I
         -FCC1254
GPO                BX6    -X1*X4                                                     GPO        4554         I
         -FCC1254
GPO                SA6    A7+B1       HA = HA + 1;  [HA] = ^DMASK & D                GPO        4555         I
         -FCC1254
GPO                SX7    B6                                                         GPO        4556         I
         -FCC1254
GPO                BX7    X5+X7       HBP[LI] = HBI                                  GPO        4557         I
         -FCC1254
GPO                SA7    A5                                                         GPO        4558         I
         -FCC1254
GPO                SB6    B6+4        HBI = HBI + 4                                  GPO        4559         I
         -FCC1254
GPO                MX6    0                                                          GPO        4560         I
         -FCC1254
GPO                SA6    A6+B1       HA = HA + 1;  [HA] = 0                         GPO        4561         I
         -FCC1254
GPO       MIE16    RJ     MIH         MOVE INSTRUCTIONS TO *HB*                      FCC1254      52    A
GPO                LX4    59-D.INCP                                                  GPO        4562    A
GPO                MI     X4,MIE14A   IF INC[D]                                      GPO        4563    A
GPO                LX7    59-I.TIP                                                   GPO        4564    A
GPO                PL     X7,MIE15    IF ^TI[LI]    */ NOT TERMINAL                  GPO        4565    A
GPO                LX4    D.INCP-D.ZPP                                               GPO        4566    A
GPO                MI     X4,MIE15    IF ZP[D]      */ NO PRED                       GPO        4567    A
GPO                                                                                  GPO        4568    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                 LX7    1+I.TIP                                          GPO         4569     A
GPO                 SX7    X7          R = HBP[LI]   */ OPERAND OF *XMT*    GPO         4570     A
GPO                 SX6    B2-OC.DEF                                        FCC1254       53     A
GPO                 ZR     X6,MIE19    IF OC[R1] = OC.DEF                   FCC1254       54     A
GPO                 LX4    D.ZPP-D.PRSP                                     GPO         4571     A
GPO                 PL     X4,MIE18    IF ^PRS[DI]                          GPO         4572     A
GPO                                                                         GPO         4573     A
GPO       *         TERMINAL AND PRS, CHECK FOR TEMP LOCK  *RS* TO A B-REG  GPO         4574     A
GPO       *         IF THE ABOVE, THEN MAKE PRED TERMINAL IF COMPUTATION IS TO BE  GPO   4575     A
GPO       *         REMOVED FROM *BLK*. EXAMPLE J = SHIFT(INV,I+5)+SHIFT(VAR,I+5)  GPO   4576     A
GPO                                                                         GPO         4577     A
GPO                 SA3    A5+B1       R1RS = LI + 1                        GPO         4578     A
GPO                 MX6    -SO.LKL                                          GPO         4579          I
          -FCC1530X
GPO                 LX3    -R1.SOP-SO.LKP                                   GPO         4580          I
          -FCC1530X
GPO                 BX6    -X6*X3                                           GPO         4581          I
          -FCC1530X
GPO                 SB2    X6                                               GPO         4582          I
          -FCC1530X
GPO                 NE     B2,B1,MIE18 IF SOLK[R1RS] " 1   */ NOT A TEMP LOCK  GPO      4583          I
          -FCC1530X
GPO                 LX4    1+D.PRSP+58-D.TYP                                GPO         4584          I
          -FCC1530X
GPO                 LX3    -R1.SOP-SO.LKP-1                                 FCC1530X       5     A
GPO                 MI     X3,MIE16A   IF SOLK[R1RS] = 1   */ A TEMP LOCK   FCC1530X       6     A
GPO                                                                         FCC1530X       7     A
GPO                 SA3    A5+4        LIRS = LI + 4                        FCC1530X       8     A
GPO                 LX3    59-I.INVP                                        FCC1530X       9     A
GPO                 PL     X3,MIE18    IF ^INV[LIRS] */ RS STAYS IN THE LOOP FCC1530X      10     A
GPO                                                                         FCC1530X      11     A
GPO                 MX2    0           LINV = 0                             FCC1530X      12     A
GPO                 EQ     MIE17A                                           FCC1530X      13     A
GPO                                                                         FCC1530X      14     A
GPO       MIE16A    LX4    1+D.PRSP+58-D.TYP                                FCC1530X      15     A
GPO                 LX2    -R1.RJP     I = RJ[R1]                           GPO         4585     A
GPO                 PL     X4,MIE17    IF TYPE[DI] = I                      GPO         4586     A
GPO                 SA2    A2+B1       R2 = R1 + 1                          GPO         4587     A
GPO                 LX2    -IH.RFP     I = RF[R2]                           GPO         4588     A
GPO       MIE17     BX6    "RN"X2                                           GPO         4589     A
GPO                 SA3    "TB"+X6     LIP = TB + I                         GPO         4590     A
GPO                 SB3    X6                                               GPO         4591     A
GPO                 SA2    A3-B1       DIP = LIP - 1                        GPO         4592     A
GPO                 SX7    X3          R = HBP[LIP]                         GPO         4593     A
GPO                 LX3    59-I.TIP                                         GPO         4594     A
GPO                 MI     X3,MIE15    IF TI[LIP]    */ PRED IS TERMINAL    GPO         4595     A
GPO                 LX2    59-D.ZPP                                         GPO         4596     A
GPO                 MI     X2,MIE15    IF ZP[DIP]                           GPO         4597     A
GPO                                                                         GPO         4598     A
GPO       *         CHAIN FOWARD AND MOVE THE RS OUT SO IT DOESNT GET SEPERATED  GPO   4599     A
GPO                                                                         GPO         4600     A
GPO                 LX3    1+I.TIP                                          GPO         4601     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                MX2    1                                                GPO          4602     A
GPO                LX2    1+I.INVP                                         GPO          4603     A
GPO                BX2    X2*X3        LINV = INV[LIP]                     GPO          4604     A
GPO                LX5    -I.IICP                                          GPO          4605          I
          -FCC1530X
GPO        MIE17A  LX5    -I.IICP                                          FCC1530X       16     A
GPO                SA5    "TB"+X5      LI = TB + IIC[LI]                   GPO          4606     A
GPO                SA4    A5-B1        DI = LI - 1                         GPO          4607     A
GPO                SA3    A5-3         R1 = LI - 3                         GPO          4608     A
GPO                BX6    X3                                               GPO          4609     A
GPO                SA6    A6+B1        HA = HA + 1;  [HA] = [R1]           GPO          4610     A
GPO                MX6    0                                                GPO          4611     A
GPO                SA6    A6+B1        HA = HA + 1;  [HA] = 0              GPO          4612     A
GPO                BX6    X4                                               GPO          4613     A
GPO                SA6    A6+B1        HA = HA + 1;  [HA] = [DI]           GPO          4614     A
GPO                MX6    0                                                GPO          4615     A
GPO                SA6    A6+B1        HA = HA + 1;  [HA] = 0              GPO          4616     A
GPO                BX5    X2+X5        INV[LI] = LINV                      GPO          4617     A
GPO                SB6    B6+4         HBI = HBI + 4                       GPO          4618     A
GPO                                                                       GPO          4619     A
GPO       *        CREATE *XMT* / *TST* TO SAVE RESULT IN A TEMPORARY     GPO          4620     A
GPO       *        LINK WORD OF *TST* POINTS TO WHERE TO PLACE *TLD*      GPO          4621     A
GPO                                                                       GPO          4622     A
GPO        MIE18   SB2    OC.XMT                                           GPO          4623     A
GPO                RJ     OST          OUTPUT XMT TST TO IT.               FCC1254        55     A
GPO                EQ     MIE15                                            FCC1254        56     A
GPO                                                                       FCC1254        57     A
GPO       *        TERMINAL *DEF* MOVED TO *HB*, MOVE ALL FOLLOWING DEF"S TO  FCC1254      58     A
GPO       *        THE *HB* BEFORE OUTPUTTING THE *TST*"S.                FCC1254        59     A
GPO                                                                       FCC1254        60     A
GPO        MIE19   SX7    B3                                               FCC1254        61     A
GPO                SA7    ISV          ISV = I                             FCC1254        62     A
GPO                                                                       FCC1254        63     A
GPO        MIE19A  LX5    -I.IICP                                          FCC1254        64     A
GPO                SB3    X5-3         I = IIC[LI] - 3                     FCC1254        65     A
GPO                SA2    "TB"+B3      R1 = TB + I                         FCC1254        66     A
GPO                UX6    B2,X2                                            FCC1254        67     A
GPO                SB3    X5           I = IIC[LI]                         FCC1254        68     A
GPO                SX7    B2-OC.DEF                                        FCC1254        69     A
GPO                NZ     X7,MIE19B    IF OC[R1] " OC.DEF                  FCC1254        70     A
GPO                RJ     MIH          MOVE INST TO *HB*                   FCC1254        71     A
GPO                EQ     MIE19A                                           FCC1254        72     A
GPO                                                                       FCC1254        73     A
GPO       *        NOW OUTPUT THE XMT/SA AND TST INSTRUCTIONS             FCC1254        74     A
GPO                                                                       FCC1254        75     A
GPO        MIE19B  SA2    ISV                                              FCC1254        76     A
GPO                SB3    X2           I = ISV                             FCC1254        77     A
GPO                                                                       FCC1254        78     A
GPO        MIE19C  SA5    "TB"+B3      LI = TB + I                         FCC1254        79     A
GPO                SA2    A5-3         R1 = LI - 3                         FCC1254        80     A
GPO                UX7    B2,X2                                            FCC1254        81     A
GPO                SX6    B2-OC.DEF                                        FCC1254        82     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                NZ     X6,MIE19F   IF OC[R1] " OC.DEF                          FCC1254      83     A
GPO                                                                               FCC1254      84     A
GPO                BX6    X5                                                      FCC1254      85     A
GPO                LX6    59-I.TIP                                                FCC1254      86     A
GPO                PL     X6,MIE19E   IF ^TI[LI]    */ NOT TERMINAL               FCC1254      87     A
GPO                                                                               FCC1254      88     A
GPO                SB2    OC.XMT      OC = OC.XMT                                 FCC1254      89     A
GPO                LX2    59-R1.SOP-4                                             FCC1254      90     A
GPO                MI     X2,MIE19D   IF REG[R1] \ 20B  */ DEF FROM AN X-REG      FCC1254      91     A
GPO                SB2    OC.SA       OC = OC.SA                                  FCC1254      92     A
GPO       MIE19D   SX7    X5          RJ = HBP[LI]                                FCC1254      93     A
GPO                RJ     OST         OUTPUT XMT/SA AND TST                       FCC1254      94     A
GPO       MIE19E   LX5    -I.IICP                                                 FCC1254      95     A
GPO                SB3    X5          I = IIC[LI]                                 FCC1254      96     A
GPO                EQ     MIE19C                                                  FCC1254      97     A
GPO                                                                               FCC1254      98     A
GPO       MIE19F   NZ     B3,MIE16    IF I " 0      */ NOT END OF CHAIN           FCC1254      99     A
GPO                EQ     MIE20                                                   FCC1254     100     A
GPO       MIE      SPACE  2,14                                                    FCC1254     101     A
GPO      **        MIH - MOVE INSTRUCTION TO HOLDING BLOCK                        FCC1254     102     A
GPO      *                                                                        FCC1254     103     A
GPO      *         ENTRY  (B3) = I                                                FCC1254     104     A
GPO      *         EXIT   B2,X2,X3,X4,X5,X7 SET                                   FCC1254     105     A
GPO                                                                               FCC1254     106     A
GPO       MIH      ROUTINE                                                        FCC1254     107     A
GPO                SA5    "TB"+B3     LI = TB + I                                 FCC1254     108     A
GPO                SA4    A5-B1       DI = LI = 1; D = [DI]                       FCC1254     109     A
GPO                SA3    A4-B1       R2 = DI - 1                                 FCC1254     110     A
GPO                SA2    A3-B1       R1 = R2 - 1                                 FCC1254     111     A
GPO                BX6    X2                                                      FCC1254     112     A
GPO                SA6    B5+B6       HA = HB + HBI;  [HA] = [R1]                 FCC1254     113     A
GPO                LX7    X3                                                      FCC1254     114     A
GPO                SA7    A6+B1       HA = HA + 1;  [HA] = [R2]                   FCC1254     115     A
GPO                BX6    -X1*X4                                                  FCC1254     116     A
GPO                SA6    A7+B1       HA = HA + 1;  [HA] = ^DMASK & D             FCC1254     117     A
GPO                SX7    B6                                                      FCC1254     118     A
GPO                BX7    X5+X7       HBP[LI] = HBI                               FCC1254     119     A
GPO                SA7    A5                                                      FCC1254     120     A
GPO                SB6    B6+4        HBI = HBI + 4                               FCC1254     121     A
GPO                MX6    0                                                       FCC1254     122     A
GPO                SA6    A6+B1       HA = HA + 1;  [HA] = 0                      FCC1254     123     A
GPO                UX2    B2,X2                                                   FCC1254     124     A
GPO                EQ     MIH                                                     FCC1254     125     A
GPO                                                                               FCC1254     126     A
GPO      **        OST - OUTPUT XMT AND TST TO IT.                                FCC1254     127     A
GPO      *                                                                        FCC1254     128     A
GPO      *         ENTRY  (B2,X7) = OC, RJ                                        FCC1254     129     A
GPO                                                                               FCC1254     130     A
GPO       OST      ROUTINE                                                        FCC1254     131     A
GPO                LX7    R1.RJP                                                  GPO        4624     A
GPO                SX4    B6                                                      GPO        4625     A
GPO                PX3    B2,X4                                                   GPO        4626     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
      GPO                BX6    X7+X3                                                GPO          4627     A
      GPO                SA6    A6+B1       HA = HA + 1;  [HA] = TYI(OC.XMT,R,0,HBI)  GPO          4628     A
      GPO                MX6    0                                                    GPO          4629     A
      GPO                SA6    A6+B1       HA + HA + 1;  [HA] = 0                   GPO          4630     A
      GPO                SA6    A6+B1       HA + HA + 1;  [HA] = 0                   GPO          4631     A
      GPO                SB2    OC.TST                                               GPO          4632     A
      GPO                SA6    A6+B1       HA + HA + 1;  [HA] = 0                   GPO          4633     A
      GPO                SA3    =XIT.                                                GPO          4634     A
      GPO                PX6    B2,X4                                                GPO          4635     A
      GPO                SA6    A6+B1       HA = HA + 1;  [HA] = TYIII(OC.TST,0,0,HBI) GPO         4636     A
      GPO                SX2    B4                                                   GPO          4637     A
      GPO                LX2    IH.CAP                                               GPO          4638     A
      GPO                LX3    IH.IHP                                               GPO          4639     A
      GPO                BX6    X2+X3                                                GPO          4640     A
      GPO                SA6    A6+B1       HA = HA + 1;  [HA] = IHW(0,TO,IT.)       GPO          4641     A
      GPO                SB4    B4+B1       TO = TO + 1                              GPO          4642     A
      GPO                MX6    0                                                    GPO          4643     A
      GPO                SA6    A6+B1       HA = HA + 1;  [HA] = 0                   GPO          4644     A
      GPO                SX7    B3                                                   GPO          4645     A
      GPO                LX7    T.BIP                                                GPO          4646     A
      GPO                SX3    A0                                                   GPO          4647     A
      GPO                BX6    X7+X3                                                GPO          4648     A
      GPO                MX3    1                                                    GPO          4649     A
      GPO                LX3    1+I.INVP                                             GPO          4650     A
      GPO                BX2    X3*X5                                                GPO          4651     A
      GPO                IX6    X2+X6                                                GPO          4652     A
      GPO                SA6    A6+B1       HA = HA + 1;  [HA] = TETW(INV,0,I,TSC)   GPO          4653          I
           -CCG21
      GPO                SA4    A5-B1       DI = LI - 1                              CCG21          22     A
      GPO                LX3    D.RFP-I.INVP                                         CCG21          23     A
      GPO                BX2    X3*X4                                                CCG21          24     A
      GPO                LX2    T.RFP-D.RFP                                          CCG21          25     A
      GPO                BX6    X2+X6                                                CCG21          26     A
      GPO                SA6    A6+B1       HA = HA + 1;  [HA] = TETW(INV,RF,0,I,TSC) CCG21         27     A
      GPO                SA0    A6-B5       TSC = HA - HB                            GPO          4654     A
      GPO                SB6    A0+B1       HBI = TSC + 1  */ = HBI + 8              GPO          4655     A
      GPO                EQ     MIE15                                                GPO          4656          I
           -FCC1254
      GPO                EQ     OST                                                  FCC1254       132     A
      GPO       MIE      SPACE  3,14                                                 GPO          4657     A
      GPO      *         RENUMBER THE INSTRUCTIONS ADDED TO THE HOLDING BLOCK        GPO          4658     A
      GPO      *         PRESERVE A0, B4 FOR *TET* UPDATE                            GPO          4659     A
      GPO                                                                            GPO          4660     A
      GPO       MIE20    MX7    0           [HB+HBI] = 0  */ SET HB TERMINATOR       GPO          4661     A
      GPO                SA7    B5+B6                                                GPO          4662     A
      GPO                SX6    B6          L.TXT = HBI                              GPO          4663     A
      GPO                SB2    B1+B1                                                GPO          4664     A
      GPO                SA6    L.TXT                                                GPO          4665     A
      GPO                MX1    D.TYL                                                GPO          4666     A
      GPO                SA2    TBB                                                  GPO          4667     A
      GPO                SA4    X2-2        DI = TBB - 2                             GPO          4668     A
      GPO                EQ     MIE22                                                GPO          4669     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                                                                    GPO        4670    A
GPO       MIE21   SA3    =XF.RDT+B3                                    GPO        4671    A
GPO               BX6    X3          [DI] = RDT(OC[R1])                GPO        4672    A
GPO               SA6    A4                                            GPO        4673    A
GPO                                                                    GPO        4674    A
GPO       MIE22   SA5    A4+B2       R1 = DI + 2                       GPO        4675    A
GPO               SA4    A5+B2       DI = R1 + 2                       GPO        4676    A
GPO               ZR     X5,MIE26    IF [R1] = 0   */ END OF BLOCK     GPO        4677    A
GPO               LX4    58-D.TYP                                      GPO        4678    A
GPO               BX6    -X1+X4                                        GPO        4679    A
GPO               UX7    B3,X5                                         GPO        4680    A
GPO               ZR     X4,MIE21    IF [DI] = 0   */ XMT/TST          GPO        4681    A
GPO               ZR     X6,MIE22    IF TYPE[DI] = IV                  GPO        4682    A
GPO                                                                    GPO        4683    A
GPO               BX7    "RN"X5                                        GPO        4684    A
GPO               SA3    "TB"+X7                                       GPO        4685    A
GPO               BX5    X0*X5                                         GPO        4686    A
GPO               SX2    X3                                            GPO        4687    A
GPO               BX4    X1*X4                                         GPO        4688    A
GPO               IX6    X2+X5       RI[R1] = HBP[TB+RI[R1]]           GPO        4689    A
GPO               NZ     X4,MIE25    IF TYPE[DI] " I                   GPO        4690    A
GPO                                                                    GPO        4691    A
GPO               LX6    -R1.RJP                                       GPO        4692    A
GPO               BX7    "RN"X6                                        GPO        4693    A
GPO               ZR     X7,MIE23    IF RJ[R1] = 0                     GPO        4694    A
GPO               SA3    "TB"+X7                                       GPO        4695    A
GPO               BX6    X0*X6                                         GPO        4696    A
GPO               SX2    X3          RJ[R1] = HBP[TB+RJ[R1]]           GPO        4697    A
GPO               BX6    X2+X6                                         GPO        4698    A
GPO       MIE23   LX6    R1.RJP-R1.RKP                                 GPO        4699    A
GPO               BX7    "RN"X6                                        GPO        4700    A
GPO               ZR     X7,MIE24    IF RK[R1] = 0                     GPO        4701    A
GPO               SA3    "TB"+X7                                       GPO        4702    A
GPO               BX6    X0*X6                                         GPO        4703    A
GPO               SX2    X3          RK[R1] = HBP[TB+RK[R1]]           GPO        4704    A
GPO               BX6    X2+X6                                         GPO        4705    A
GPO       MIE24   LX6    R1.RKP                                        GPO        4706    A
GPO               SA6    A5                                            GPO        4707    A
GPO               EQ     MIE22                                         GPO        4708    A
GPO                                                                    GPO        4709    A
GPO       MIE25   SA3    A5+B1       R2 = R1 + 1                       GPO        4710    A
GPO               SA6    A5                                            GPO        4711    A
GPO               LX3    -IH.RFP                                       GPO        4712    A
GPO               BX7    "RN"X3                                        GPO        4713    A
GPO               ZR     X7,MIE22    IF RF[R2] @ 1                     GPO        4714    A
GPO               SA2    "TB"+X7                                       GPO        4715    A
GPO               BX3    X0*X3                                         GPO        4716    A
GPO               SX7    X2                                            GPO        4717    A
GPO               IX6    X3+X7                                         GPO        4718    A
GPO               LX6    IH.RFP      RF[R2] = HBP[TB+RF[R2]]           GPO        4719    A
GPO               SA6    A3                                            GPO        4720    A
GPO               EQ     MIE22                                         GPO        4721    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO       MIE      SPACE  3,20                                                    GPO        4722    A
GPO      *         RESCAN THE CHAIN OF MOVED INSTRUCTIONS AND CHANGE THE ONES      GPO        4723    A
GPO      *         THAT ARE ^D AND NOT A TEMP LOCK *RS* TO *BOS*"S SO *SQZ*        GPO        4724    A
GPO      *         WILL ELIMINATE THEM WHEN THE BLOCK IS SQUEEZED AGAIN.           GPO        4725    A
GPO                                                                                GPO        4726    A
GPO       MIE26    SA1    IC          I = IC                                       GPO        4727    A
GPO                SA2    "TB"-B1                                                  GPO        4728    A
GPO                SB2    B1+B1                                                    GPO        4729    A
GPO                SA3    A2-B2                                                    GPO        4730    A
GPO                MX0    -SO.LKL                                                  GPO        4731    A
GPO                LX7    X2                                                       GPO        4732    A
GPO                BX6    X3                                                       GPO        4733    A
GPO                SB5    OC.RS                                                    GPO        4734    A
GPO                LX0    R1.SOP+SO.LKP                                            GPO        4735    A
GPO                                                                                GPO        4736    A
GPO       MIE27    SA5    "TB"+X1     LI = TB + I                                  GPO        4737    A
GPO                SA4    A5-B1       DI = LI + 1                                  GPO        4738    A
GPO                LX5    -I.IICP                                                  GPO        4739    A
GPO                LX4    59-D.^DP                                                 GPO        4740    A
GPO                SX1    X5          I = IIC[LI]   */ NEXT ON CHAIN               GPO        4741    A
GPO                PL     X4,MIE29    IF DEF[DI]                                   GPO        4742    A
GPO                                                                                GPO        4743    A
GPO                SA2    A4-B2       R1 = DI - 2                                  GPO        4744    A
GPO                UX3    B3,X2                                                    GPO        4745    A
GPO                NE     B3,B5,MIE28 IF OC[R1] " OC.RS                            GPO        4746    A
GPO                BX4    -X0*X2                                                   GPO        4747    A
GPO                NZ     X4,MIE29    IF SOLK[R1] " 0  */ TEMP LOCK                GPO        4748    A
GPO       MIE28    SA6    A4-2        [DI+2] = TYI(OC.BOS,0,XX)                    GPO        4749    A
GPO                SA7    A4+         [DI] = [TB-1]                                GPO        4750    A
GPO                                                                                GPO        4751    A
GPO       MIE29    NZ     X1,MIE27    IF I " 0                                     GPO        4752    A
GPO                                                                                GPO        4753    A
GPO                TRACE  MIE,RLIST                                                GPO        4754    A
GPO                TRACE  MIE,RLIST,SEQ                                            GPO        4755    A
GPO       MIE      SPACE  3,14                                                     GPO        4756    A
GPO      *         MOVE THE INFO IN THE LINK WORD OF THE *TST*"S TO *TET*          GPO        4757    A
GPO      *         SET NTP " 0 IF TERMINAL POLYNOMIALS WERE MOVED TO HOLDING BLK.  GPO        4758    A
GPO                                                                                GPO        4759    A
GPO                SA2    L.TET                                                    GPO        4760    A
GPO                SX0    A0          I = TSC       */ TSC CHAIN INDEX             GPO        4761    A
GPO                SX5    B4                                                       GPO        4762    A
GPO                IX1    X5-X2                                                    GPO        4763    A
GPO                SA4    BSW                                                      GPO        4764    A
GPO       .BMB     BIT    BS.BMP-BS.NTETP                                          GPO        4765    A
GPO                SX2    X1+.BMB     BM[BSW] = 1   */  INDCIATE BLOCK MODIFIED    GPO        4766    A
GPO                LX2    BS.NTETP                                                 GPO        4767    A
GPO                BX6    X2+X4       NTET[BSW] = T0 - L.TET                       GPO        4768    A
GPO                SA6    A4                                                       GPO        4769    A
GPO                ZR     X1,MIE      IF TO = L.TET */ NO TEMP STORES              GPO        4770    A
GPO                                                                                GPO        4771    A
GPO                ALLOC  TET,X1      ALLOC( TET , TO - L.TET )                    GPO        4772    A
GPO                SA4    O.TXT                                                    GPO        4773    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                SB5    X2                                                GPO          4774     A
GPO                SB2    B1+B1                                             GPO          4775     A
GPO                MX1    -T.CAL                                            GPO          4776     A
GPO                SA2    NTP                                               GPO          4777     A
GPO                SB4    X4          HB = O.TXT                            GPO          4778     A
GPO                BX7    X7-X7                                             GPO          4779     A
GPO                MX3    1                                                 GPO          4780     A
GPO                LX3    1+I.INVP                                          GPO          4781     A
GPO                                                                         GPO          4782     A
GPO       MIE30    SA5    B4+X0       LI = HB + I                           GPO          4783     A
GPO                SA4    A5-B2       R2 = LI - 2                           GPO          4784     A
GPO                BX6    -X5*X3                                            GPO          4785     A
GPO                LX4    -IH.CAP                                           GPO          4786     A
GPO                BX2    X6+X2       NTP = ^INV[LI] ! NTP                  GPO          4787     A
GPO                SX0    X5          I = LINK[LI]                          GPO          4788     A
GPO                BX6    X1*X5                                             GPO          4789     A
GPO                SA6    B5+X4       TET(CA[R2]) = TETW(INV,0,BI,0)        GPO          4790          I
          -CCG21
GPO                SA6    B5+X4       TET(CA[R2]) = TETW(INV,RF,0,BI,0)     CCG21          28     A
GPO                SA7    A5          [LI] = 0                              GPO          4791     A
GPO                NZ     X0,MIE30    IF I " 0                              GPO          4792     A
GPO                                                                         GPO          4793     A
GPO                BX6    X2                                                GPO          4794     A
GPO                SA6    A2                                                GPO          4795     A
GPO                TRACE  MIE,TET                                           GPO          4796     A
GPO                EQ     MIE                                               GPO          4797     A
GPO       CIF      TITLE  CIF - COLLECT INCREMENT FORMULAS                  GPO          4798     A
GPO      **        CIF - COLLECT INCREMENT FORMULAS                         GPO          4799     A
GPO      *                                                                  GPO          4800     A
GPO      *         BUILD A TABLE OF INCREMENT AND POLYNOMIAL FORMULAS OF THE GPO          4801     A
GPO      *         OF THE *IP*"S ( *IIT* LIST B FORMAT ).                   GPO          4802     A
GPO      *         THE INCREMENT FORMULAS ARE USED TO EVALUATE THE INCREMENT GPO          4803     A
GPO      *         EXPRESSIONS. THE POLYNOMIAL FORMULAS ARE NECESSARY FOR TEST GPO        4804     A
GPO      *         REPLACEMENT AND TO COMPUTE THE SYMBOLIC DIFFERENCE OF TWO GPO          4805     A
GPO      *         SIMILAR *IP*"S ( A(I,J+1) AND A(I,J) , VARIABLE DIMENSIONS ). GPO      4806     A
GPO                                                                         GPO          4807     A
GPO       CIF0     SX6    0                                                 FCC1710X        8     A
GPO                SA6    NTP         NTP = 0                               FCC1710X        9     A
GPO       CIF      ROUTINE                                                  GPO          4808     A
GPO                                                                         GPO          4809     A
GPO      *         CALL *MII* TO MARK THE *INV* INSTRUCTIONS IN THE *HB* FOR DIF. GPO     4810     A
GPO                                                                         GPO          4811     A
GPO                MX4    60          BSW = -0                              GPO          4812     A
GPO                SA5    O.TXT                                             GPO          4813     A
GPO                SX6    B0                                                GPO          4814     A
GPO                MX0    -R1.RIL+1                                         GPO          4815          I
          -CCG11
GPO                SA6    BN          BN = 0        */ INDICATE SPECIAL CALL GPO          4816     A
GPO                LX0    1                                                 GPO          4817          I
          -CCG11
GPO                SX0    RN.MASK                                           CCG11         119     A
GPO                RJ     MII         MARK INVARIANT INSTRUCTIONS           GPO          4818     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                                                                                GPO        4819    A
GPO     *         SCAN HOLDING BLOCK TO FORM A CHAIN OF *TST*"S TO *TP*"S.         GPO        4820    A
GPO                                                                                GPO        4821    A
GPO               SA1    L.TXT                                                     GPO        4822    A
GPO               SA2    O.TXT                                                     GPO        4823    A
GPO               SB3    OC.TST                                                    GPO        4824    A
GPO               SB4    4                                                         GPO        4825    A
GPO               S"TB"  X2+3        TB = O.TXT + 3                                GPO        4826    A
GPO               MX7    0           SC = 0                                        GPO        4827    A
GPO               IX6    X1+X2                                                     GPO        4828    A
GPO               SA5    X6-4        R1 = O.TXT + L.TXT-4                          GPO        4829    A
GPO               AX1    2                                                         GPO        4830    A
GPO               SA2    X2          R1 = [O.TXT]                                  GPO        4831    A
GPO               MX6    -R1.H2L                                                   GPO        4832    A
GPO               LX6    R1.H2P                                                    GPO        4833    A
GPO               BX6    X6*X2       H2[R1] = 0                                    GPO        4834    A
GPO               SA6    A2                                                        GPO        4835    A
GPO               SA3    O.TET                                                     FCC1710X     10    A
GPO               MX2    1                                                         FCC1710X     11    A
GPO               LX2    1+T.INVP                                                  FCC1710X     12    A
GPO               SB5    X3                        (B5) = TET                      FCC1710X     13    A
GPO                                                                                GPO        4836    A
GPO       CIF1    SA5    A5-B4       R1 = R1 - 4                                   GPO        4837    A
GPO               UX6    B2,X5                                                     GPO        4838    A
GPO               EQ     B2,B1,CIF2  IF OC[R1] = OC.BOS                            GPO        4839    A
GPO               NE     B2,B3,CIF1  IF OC[R1] " OC.TST                            GPO        4840    A
GPO               SA4    A5+3        LI = R1 + 3                                   GPO        4841    A
GPO               LX4    59-I.INVP                                                 GPO        4842    A
GPO               MI     X4,CIF1     IF INV[LI]                                    GPO        4843    A
GPO               SA3    "TB"+X5     LIP = TB + RI[R1]                             FCC1710X     14    A
GPO               LX3    59-I.INVP                                                 FCC1710X     15    A
GPO               MI     X3,CIF1A    IF INV[LIP]                                   FCC1710X     16    A
GPO               SA7    A4          [LI] = SC                                     GPO        4844    A
GPO               SX7    A4-"TB"     SC = LI - TB                                  GPO        4845    A
GPO               EQ     CIF1                                                      GPO        4846    A
GPO                                                                                FCC1710X     17    A
GPO     *         TST IS *IP*, BUT EXPRESSION IS *INV* UPDATE *TET* TO REFLECT     FCC1710X     18    A
GPO     *         THE REDUCTION OF THE EXPRESSION ( I+INVEXP-I, ETC ) BY *SQZ* .   FCC1710X     19    A
GPO                                                                                FCC1710X     20    A
GPO      CIF1A    SA3    A5+B1       R2 = R1 + 1                                   FCC1710X     21    A
GPO               LX3    -IH.CAP                                                   FCC1710X     22    A
GPO               SA4    B5+X3                                                     FCC1710X     23    A
GPO               BX6    X2+X4       INV[TET(CA[R2])] = 1                          FCC1710X     24    A
GPO               SA6    A4                                                        FCC1710X     25    A
GPO               EQ     CIF1                                                      FCC1710X     26    A
GPO                                                                                GPO        4847    A
GPO      CIF2     SA7    TSC         TSC = SC                                      GPO        4848    A
GPO               ZR     X7,CIF0     IF SC = 0     */ NO TERMINAL POLYNOMIALS      FCC1710X     27    A
GPO               ALLOC  RND,X1      ALLOC( RND , L.TXT/4 )                        GPO        4849    A
GPO      CIF      SPACE  1,10                                                      GPO        4850    A
GPO     *         FORM A LIST OF THE INCREMENT AND POLYNOMIAL FORMULAS IN *IIT*    GPO        4851    A
GPO                                                                                GPO        4852    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                SA5    TSC                                                GPO        4853    A
GPO                ADDWRD IIT,B0      ADDWRD( IIT , 0 )                      GPO        4854    A
GPO                SX6    X3-1                                               GPO        4855    A
GPO                SA6    ITB         ITB = L.IIT - 1  */ INDEX TO *PI* OF THIS GPO        4856    A
GPO                                                                           GPO        4857    A
GPO       CIF3     SA4    O.TXT                                              GPO        4858    A
GPO                IX7    X4+X5                                              GPO        4859    A
GPO                SA5    X7+B1       R2 = TXT(TSC+1)                        GPO        4860    A
GPO                LX5    -IH.CAP                                            GPO        4861    A
GPO                SX6    X5          CAS = CA[R2]  */ SAVE *CA* OF THIS *IP* GPO        4862    A
GPO                SA6    CAS                                                GPO        4863    A
GPO                SB6    0           MULT = 0                               GPO        4864    A
GPO                RJ     DIF         DEVELOP *INCR* FORMULA                 GPO        4865    A
GPO                SA1    LCV                                                GPO        4866    A
GPO                SA5    A0                                                 GPO        4867    A
GPO                LX5    -IA.UDIP                                           GPO        4868    A
GPO                UX6    B2,X1                                              GPO        4869    A
GPO                AX6    PS.UDIP                                            GPO        4870    A
GPO                BX7    X6-X5                                              GPO        4871    A
GPO                SB2    X7+                                                GPO        4872    A
GPO                NZ     B2,CIF4     IF UDI[LCV] " UDI[IAW]                 GPO        4873    A
GPO                SA6    STP         STP = 1       */ AN *IP* IS A FUNCTION(LCV) GPO        4874    A
GPO                                                                           GPO        4875    A
GPO       CIF4     SA1    TSC                                                GPO        4876    A
GPO                SA4    O.TXT                                              GPO        4877    A
GPO                IX7    X4+X1                                              GPO        4878    A
GPO                SA5    X7+B1       R2 = TXT(TSC+1)                        GPO        4879    A
GPO                SA3    X7+3                                               GPO        4880    A
GPO                BX7    X3          TSC = TXT(TSC+3)  */ INDEX TO NEXT ON CHAIN GPO        4881    A
GPO                SA7    A1                                                 GPO        4882    A
GPO                SB6    -1          MULT = -1                              GPO        4883    A
GPO                RJ     DIF         DEVELOP *POLYNOMIAL* FORMULA           GPO        4884    A
GPO                ADDWRD IIT,B0      ADDWRD( IIT , 0 )                      GPO        4885    A
GPO                                                                           GPO        4886    A
GPO                SA1    ITB                                                GPO        4887    A
GPO                SA4    CAS                                                GPO        4888    A
GPO                SB2    X2                                                 GPO        4889    A
GPO                SA2    O.TET                                              GPO        4890    A
GPO                SB4    X4                                                 GPO        4891    A
GPO                SX6    X3-1                                               GPO        4892    A
GPO                LX6    PI.LFP                                             GPO        4893    A
GPO                SA6    B2+X1       IIT(ITB) = L.IIT - 1  */ LINK TO NEXT  GPO        4894    A
GPO                LX1    T.ITIP                                             GPO        4895    A
GPO                SA2    X2+B4                                              GPO        4896    A
GPO                BX7    X1+X2       ITI[TET(CAS)] = ITB  */ LINK *IP* TO *IIT* GPO        4897    A
GPO                SA7    A2                                                 GPO        4898    A
GPO                LX6    -PI.LFP                                            GPO        4899    A
GPO                SA6    A1          ITB = L.IIT - 1                        GPO        4900    A
GPO                                                                           GPO        4901    A
GPO                SA5    TSC                                                GPO        4902    A
GPO                NZ     X5,CIF3     IF TSC " 0                             GPO        4903    A
GPO        CIF     SPACE  1,10                                               GPO        4904    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO      *         NOW SCAN THE *PI* LIST FOR SIMILAR *IP*"S, THOSE WHICH HAVE      GPO        4905    A
GPO      *         THE SAME INCREMENT FORMULAS.                                     GPO        4906    A
GPO                                                                                 GPO        4907    A
GPO                SA1    O.TET                                                     GPO        4908    A
GPO                SA2    O.IIT                                                     GPO        4909    A
GPO                SA3    PIC                                                       GPO        4910    A
GPO                SB2    3                                                         GPO        4911    A
GPO                SB7    X1                        (B7) = [O.TET]                  GPO        4912    A
GPO                SA0    X2                        (A0) = [O.IIT]                  GPO        4913    A
GPO                SB5    X3          J = PIC                                       GPO        4914    A
GPO                SB6    B1          CCN = 1       */ CURRENT CLASS NUMBER         FCC0514      77    A
GPO                MX7    0                                                         FCC0514      78    A
GPO                SA7    CNI         CNI = 0                                       FCC0514      79    A
GPO                                                                                 GPO        4915    A
GPO       CIF5     SA1    A0+B5       I = J;  PIW = IIT(I)                          GPO        4916    A
GPO                ZR     X1,CIF10    IF PIW = 0    */ END OF TABLE                 GPO        4917    A
GPO                LX1    -PI.LFP                                                   GPO        4918    A
GPO                SB5    X1          J = LF[PIW]                                   GPO        4919    A
GPO                LX1    59-PI.MIPP+PI.LFP                                         GPO        4920    A
GPO                MI     X1,CIF5     IF MIP[PIW]   */ SIMILAR TO ANOTHER           GPO        4921    A
GPO                                                                                 GPO        4922    A
GPO                SA1    A1+B1       IAW = IIT(I+1)                                GPO        4923    A
GPO                UX6    B3,X1       NW = NWD[IAW]                                 GPO        4924    A
GPO                LX6    -IA.CAP                                                   GPO        4925    A
GPO                SX7    X6          CAS = CA[IAW]                                 GPO        4926    A
GPO                SA7    CAS                                                       GPO        4927    A
GPO                SX0    B1                                                        GPO        4928    A
GPO                SA2    B7+X6                                                     GPO        4929    A
GPO                LX0    T.BIPP                                                    GPO        4930    A
GPO                BX6    X0+X2       BIP[TET(CAS)] = 1  */ MARK THIS *IP* AS BAS   GPO        4931    A
GPO                SA6    A2                                                        GPO        4932    A
GPO                SB4    B5          K = J                                         GPO        4933    A
GPO                LX0    PI.MIPP-T.BIPP                                            GPO        4934    A
GPO                MX7    0                                                         FCC0514      80    A
GPO                SA2    CNI                                                       FCC0514      81    A
GPO                SB6    B6+X2       CCN = CCN + CNI  */ SET CLASS NUMBER          FCC0514      82    A
GPO                SA7    A2          CNI = 0                                       FCC0514      83    A
GPO                                                                                 GPO        4935    A
GPO      *         SEARCH REST OF LIST FOR *IP*"S THAT ARE SIMILAR                  GPO        4936    A
GPO                                                                                 GPO        4937    A
GPO       CIF7     SA2    A0+B4       PIK = IIT(K)                                  GPO        4938    A
GPO                ZR     X2,CIF5     IF PIK = 0    */ END OF LIST                  GPO        4939    A
GPO                                                                                 GPO        4940    A
GPO                SA5    A2+B1       IK = IIT(K+1)                                 GPO        4941    A
GPO                BX6    X1-X5                                                     GPO        4942    A
GPO                SX3    B3-2        N = NW - 2                                    GPO        4943    A
GPO                LX6    -IA.CAP                                                   GPO        4944    A
GPO                MX7    -IA.CAL                                                   GPO        4945    A
GPO                BX6    X7*X6                                                     GPO        4946    A
GPO                NZ     X6,CIF9     IF (NWD,UDI)[IAW] " (NWD,UDI)[IK]             GPO        4947    A
GPO                SA4    A1+B1       II = IIT(I+2)                                 GPO        4948    A
GPO                SA5    A5+B1       IK = IK + 1                                   GPO        4949    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO              BX6    X4-X5                                          GPO        4950    A
GPO              NZ     X6,CIF9     IF [II] " [IK]  */ IB WORDS  NE    GPO        4951    A
GPO              SA4    A4+B1       II = II + 1                        GPO        4952    A
GPO              SA5    A5+B1       IK = IK + 1                        GPO        4953    A
GPO                                                                   GPO        4954    A
GPO       CIF8   BX6    X4-X5       DIF = [II] - [IK]                  GPO        4955    A
GPO              SA4    A4+B1       II = II + 1                        GPO        4956    A
GPO              SA5    A5+B1       IK = IK + 1                        GPO        4957    A
GPO              BX7    X4-X5                                          GPO        4958    A
GPO              NZ     X6,CIF9     IF DIF " 0    */ R1 WORDS          GPO        4959    A
GPO              NZ     X7,CIF9     IF [II] " [IK]                     GPO        4960    A
GPO              SX3    X3-4        N = N - 4                          GPO        4961    A
GPO              SA4    A4+B2       II = II + 3                        GPO        4962    A
GPO              SA5    A5+B2       IK = IK + 3                        GPO        4963    A
GPO              NZ     X3,CIF8     IF N " 0                           GPO        4964    A
GPO                                                                   GPO        4965    A
GPO      *       *IP*"S MATCH, MARK K AS SIMILAR TO I                  GPO        4966    A
GPO                                                                   GPO        4967    A
GPO              SA3    CAS                                            GPO        4968    A
GPO              LX3    PI.CAP                                         GPO        4969    A
GPO              BX4    X0+X3       CA[IIT(K)] = CAS                   GPO        4970    A
GPO              BX6    X4+X2       MIP[IIT(K)] = 1                    GPO        4971         I
         -FCC0514
GPO              SA6    A2                                             GPO        4972         I
         -FCC0514
GPO              SA5    A1-B1                                          FCC0514      84    A
GPO              BX6    X4+X2       MIP[IIT(K)] = 1                    FCC0514      85    A
GPO              SX7    B6                                             FCC0514      86    A
GPO              LX7    PI.CNP                                         FCC0514      87    A
GPO              BX6    X7+X6       CN[IIT(K)] = CCN                   FCC0514      88    A
GPO              SA6    A2                                             FCC0514      89    A
GPO              BX7    X7+X5       CN[IIT(I)] = CCN  */ SET *CN* IN BASE  FCC0514  90    A
GPO              SA7    A5                                             FCC0514      91    A
GPO              SX6    B1                                             FCC0514      92    A
GPO              SA6    CNI         CNI = 1       */ SET FOR *CN* INCREMENT  FCC0514  93  A
GPO              SA6    NBIP        NBIP = 1      */ SET FLAG FOR *BDT*  FCC0514     94    A
GPO                                                                   FCC0514      95    A
GPO      *       ADVANCE TO NEXT *IP* IN *IIT*                         FCC0514      96    A
GPO                                                                   GPO        4973    A
GPO       CIF9   LX2    -PI.LFP                                        GPO        4974    A
GPO              SB4    X2          K = LF[PIK]                        GPO        4975    A
GPO              EQ     CIF7                                           GPO        4976    A
GPO                                                                   GPO        4977    A
GPO       CIF10  SX6    4           L.TXT = 4                          GPO        4978    A
GPO              SA6    L.TXT                                          GPO        4979    A
GPO              SX7    0                                              GPO        4980    A
GPO              SA7    L.RND       L.RND = 0                          GPO        4981    A
GPO                                                                   GPO        4982    A
GPO              IF     DEF,/DEBUG/CIF,1                               GPO        4983    A
GPO              DCALL  DMPIIT,([O.IIT],L.IIT,PIC)                     GPO        4984    A
GPO                                                                   GPO        4985    A
GPO              EQ     CIF                                            GPO        4986    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO       DIF      TITLE  DIF - DEVELOP INCREMENT FORMULAS                          GPO        4987     A
GPO      **        DIF - DEVELOP INCREMENT FORMULA OF AN *IP* IN A SINGLE VAR,      GPO        4988     A
GPO      *                OR EXTRACT AN  *IP* FORMULA.                             GPO        4989     A
GPO      *                                                                         GPO        4990     A
GPO      *         ENTRY  (B6) = MULT , =0 IF TO FORM *INC* FORMULA, ELSE = -1      GPO        4991     A
GPO      *                (X4) = [O.TXT]                                           GPO        4992     A
GPO      *                (X0) = R-NUMBER EXTRACTOR MASK                           GPO        4993     A
GPO      *                (A5) = R2 , OF *TST* OF *IP*                             GPO        4994     A
GPO      *                (CAS) = CA OF THE *IP*                                   GPO        4995     A
GPO      *                *RND* ALLOCATION MADE FOR SCRATCH SPACE                  GPO        4996     A
GPO      *                                                                         GPO        4997     A
GPO      *         EXIT   (A0) = IB , ADDRESS OF *FORMULA IN *IIT*                 GPO        4998     A
GPO                                                                                GPO        4999     A
GPO       DIF      ROUTINE                                                         GPO        5000     A
GPO                SA1    O.RND                                                    GPO        5001     A
GPO                SA2    L.RND                                                    GPO        5002     A
GPO                S"TB"  X4+3        TB = O.TXT + 3                               GPO        5003     A
GPO                IX2    X1+X2                                                    GPO        5004     A
GPO                SB5    X1                                                       GPO        5005     A
GPO                SB3    -3                                                       GPO        5006     A
GPO                SA0    X2-2        SI = O.RND + L.RND-2  */ STACK ADDRESS       GPO        5007     A
GPO                MX7    0                                                        GPO        5008     A
GPO                SA7    "TB"        [TB] = 0                                     GPO        5009     A
GPO                SA7    A0+B1       [SI+1] = 0    */ STACK TERMINATOR            GPO        5010     A
GPO                SA7    X1          OI = O.RND;  [OI] = 0  */ OUTPUT LIST        GPO        5011     A
GPO                SA5    A5-B1       R1 = R2 - 1                                  GPO        5012     A
GPO                BX6    "RN"X5                                                   GPO        5013     A
GPO                SX4    X6+B3                                                    GPO        5014     A
GPO                SA5    "TB"+X4     R1 = TB + RI[R1]-3                           GPO        5015     A
GPO                                                                               GPO        5016     A
GPO      *         PHASE 1 - TOP DOWN TO FORM INCREMENT FORMULA IN OUTPUT LIST      GPO        5017     A
GPO      *         PROCESS NEXT INSTRUCTION                                        GPO        5018     A
GPO                                                                               GPO        5019     A
GPO       DIF1     SA4    A5+2        DI = R1 + 2                                  GPO        5020     A
GPO                BX7    X5          R1W = [R1]                                   GPO        5021     A
GPO                LX4    58-D.TYP                                                 GPO        5022     A
GPO                IX3    X4+X4                                                    GPO        5023     A
GPO                MI     X4,DIF14    IF TYPE[DI] = III                            GPO        5024     A
GPO                PL     X3,DIF5     IF TYPE[DI] = I                              GPO        5025     A
GPO                                                                               GPO        5026     A
GPO      *         OUTPUT INSTRUCTION, PUSHUP STACK                                GPO        5027     A
GPO                                                                               GPO        5028     A
GPO       DIF2     SA3    A5-B3       LI = R1 + 3                                  GPO        5029     A
GPO                SX1    B1                                                       GPO        5030     A
GPO                LX1    I.MP                                                     GPO        5031     A
GPO                BX6    X1+X3                                                    GPO        5032     A
GPO                SA0    A0+B1       SI = SI + 1                                  GPO        5033     A
GPO                LX3    59-I.MP                                                  GPO        5034     A
GPO                MI     X3,DIF2A    IF M[LI]      */ ALREADY OUTPUT              GPO        5035     A
GPO                                                                               GPO        5036     A
GPO                SA7    A7+1        OI = OI + 1;  [OI] = R1W                     GPO        5037     A
GPO                SA6    A3+         M[LI] = 1     */ INDICATE IN OUTPUT LIST     GPO        5038     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                                                                    GPO        5039     A
GPO       DIF2A    SA3    A0          R1W = [SI]                       GPO        5040     A
GPO                BX2    "RN"X3                                       GPO        5041     A
GPO                ZR     X3,DIF15    IF R1W = 0    */ END OF STACK    GPO        5042     A
GPO                SB2    X2+B3                                        GPO        5043     A
GPO                SA5    "TB"+B2     R1 = TB + RI[R1]-3               GPO        5044     A
GPO                BX5    X3                                           GPO        5045     A
GPO                EQ     DIF1                                         GPO        5046     A
GPO                                                                    GPO        5047     A
GPO      *         STACK INSTRUCTION, FOLLOW SELECTED OPERAND, (X6) = OP  GPO     5048     A
GPO                                                                    GPO        5049     A
GPO       DIF4     SX4    X6+B3                                        GPO        5050     A
GPO                BX6    X7                                           GPO        5051     A
GPO                SA6    A0          [SI] = R1W; RIL = R1W            GPO        5052     A
GPO                UX7    B2,X7                                        GPO        5053     A
GPO                SA0    A0-B1       SI = SI - 1                      GPO        5054     A
GPO                SX3    B2-OC.XMT                                    GPO        5055     A
GPO                SA5    "TB"+X4     R1 = TB + OP-3; R1W = [R1]       GPO        5056     A
GPO                NZ     X3,DIF1     IF OC[R1L] " OC.XMT              GPO        5057     A
GPO                SA2    A0+2                                         GPO        5058     A
GPO                NZ     X2,DIF1     IF [SI+2] " 0 */ NOT AT TOP OF STACK  GPO  5059     A
GPO                SA0    A0+B1       SI = SI + 1;  */ ERASE XMT       GPO        5060     A
GPO                EQ     DIF1                                         GPO        5061     A
GPO                                                                    GPO        5062     A
GPO      *         PROCESS TYPE I INSTRUCTIONS                          GPO        5063     A
GPO                                                                    GPO        5064     A
GPO       DIF5     LX5    -R1.RKP                                      GPO        5065     A
GPO                BX4    "RN"X5      K = RK[R1]                       GPO        5066     A
GPO                LX5    R1.RKP-R1.RJP                                GPO        5067     A
GPO                BX6    "RN"X5      OP = J = RJ[R1]                  GPO        5068     A
GPO                NZ     X4,DIF6     IF K " 0                         GPO        5069          I
          -FCC2230X
GPO                NZ     X6,DIF5A    IF RJ[R1] " 0                    FCC2230X      7     A
GPO                BX6    X4          OP = RK[R1]                      FCC2230X      8     A
GPO                EQ     DIF5B                                        FCC2230X      9     A
GPO                                                                    FCC2230X     10     A
GPO       DIF5A    NZ     X4,DIF6     IF K " 0                         FCC2230X     11     A
GPO                                                                    GPO        5070     A
GPO      *         UNARY INSTRUCTION ( SXT, XMT, XMTC )                 GPO        5071     A
GPO      *         (PK AND NR WITH RJ = 0)                              FCC2230X     12     A
GPO                                                                    GPO        5072     A
GPO                SA4    "TB"+X6     LJ = TB + OP                     GPO        5073          I
          -FCC2230X
GPO       DIF5B    SA4    "TB"+X6     LJ = TB + OP                     FCC2230X     13     A
GPO                LX4    59-I.MP                                      GPO        5074     A
GPO                MI     X4,DIF2     IF M[LJ]      */ OPERAND PROCESSED  GPO       5075     A
GPO                EQ     DIF4                                         GPO        5076     A
GPO                                                                    GPO        5077     A
GPO       DIF6     UX3    B2,X7                                        GPO        5078     A
GPO                SX2    B2-OC.IM                                     GPO        5079     A
GPO                NZ     X2,DIF10    IF OC[R1W] " OC.IM               GPO        5080     A
GPO                                                                    GPO        5081     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO      *         PROCESS *IM*, SET MULT FLAG WHEN PROCESSING *INV* OPERAND          GPO        5082    A
GPO                                                                                   GPO        5083    A
GPO               BX2    "RN"X7                                                       GPO        5084    A
GPO               SB2    X2                                                           GPO        5085    A
GPO               NE     B2,B6,DIF7  IF RI[R1W] " MULT                                GPO        5086    A
GPO               SB6    0           MULT = 0                                         GPO        5087    A
GPO                                                                                   GPO        5088    A
GPO       DIF7    SA3    "TB"+X6     LJ = TB + OP                                     GPO        5089    A
GPO               LX3    59-I.MP                                                      GPO        5090    A
GPO               PL     X3,DIF8     IF ^M[LJ]                                        GPO        5091    A
GPO               SA3    "TB"+X4     LJ = TB + K                                      GPO        5092    A
GPO               LX3    59-I.MP                                                      GPO        5093    A
GPO               MI     X3,DIF2     IF M[LJ]      */ BOTH OPERANDS MARKED            GPO        5094    A
GPO                                                                                   GPO        5095    A
GPO       DIF8    SX6    A3-"TB"     OP = LJ - TB  */ INDEX OF OPERAND                GPO        5096    A
GPO               NZ     B6,DIF4     IF MULT " 0                                      GPO        5097    A
GPO               LX3    I.MP-I.INVP                                                  GPO        5098    A
GPO               PL     X3,DIF4     IF ^INV[LJ]                                      GPO        5099    A
GPO               SB6    B2          MULT = RI[R1W]    */ INDICATE INV OP OF IM       GPO        5100    A
GPO               EQ     DIF4                                                         GPO        5101    A
GPO       IA      SPACE  2,14                                                         GPO        5102    A
GPO      *         PROCESS IA, IS . IGNORE *INV* TERMS IF MULT = 0 .                  GPO        5103    A
GPO                                                                                   GPO        5104    A
GPO       DIF10   SA3    "TB"+X6     LJ = TB + J                                      GPO        5105         I
         -FCC0238AX
GPO       DIF10   SX2    B2-OC.UP                                                     FCC0238AX    43    A
GPO               ZR     X2,DIF13    IF OC[R1W] = OC.UP                               FCC0238AX    44    A
GPO                                                                                   FCC0238AX    45    A
GPO               SA3    "TB"+X6     LJ = TB + J                                      FCC0238AX    46    A
GPO               SA2    "TB"+X4     LK = TB + K                                      GPO        5106    A
GPO               BX1    X2*X3                                                        GPO        5107         I
         -FCC0640
GPO               LX1    59-I.MP                                                      GPO        5108         I
         -FCC0640
GPO               MI     X1,DIF2     IF M[LJ] & M[LK]                                 GPO        5109         I
         -FCC0640
GPO               NZ     B6,DIF12    IF MULT " 0                                      GPO        5110    A
GPO               BX1    X2+X3                                                        GPO        5111    A
GPO               LX1    59-I.INVP                                                    GPO        5112    A
GPO               PL     X1,DIF12    IF ^(INV[LJ] ! INV[LK])                          GPO        5113    A
GPO                                                                                   GPO        5114    A
GPO      *         ONE OPERAND IS *INV*, CHANGE INST TO AN *XMT/XMTC*                 GPO        5115    A
GPO                                                                                   GPO        5116    A
GPO               SB4    OC.XMT      OC = OC.XMT                                      GPO        5117    A
GPO               LX3    59-I.INVP                                                    GPO        5118    A
GPO               PL     X3,DIF11    IF ^INV[LJ]                                      GPO        5119    A
GPO               SB2    B2-OC.IA                                                     GPO        5120    A
GPO               SX6    X4+         J = K                                            GPO        5121    A
GPO               ZR     B2,DIF11    IF OC[R1W] = OC.IA                               GPO        5122    A
GPO               SB4    OC.XMTC     OC = OC.XMTC                                     GPO        5123    A
GPO                                                                                   GPO        5124    A
GPO       DIF11   LX6    R1.RJP                                                       GPO        5125    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO              BX1    "RN"X7                                          GPO        5126    A
GPO              PX6    B4,X6                                           GPO        5127    A
GPO              BX7    X1+X6       R1W = TYI(OC,J,0,RI[R1W])           GPO        5128    A
GPO              LX6    -R1.RJP     OP = J                              GPO        5129    A
GPO              SA3    "TB"+X6     LJ = TB + J                         FCC0640       5    A
GPO              LX3    59-I.MP                                         FCC0640       6    A
GPO              MI     X3,DIF2     IF M[LJ]                            FCC0640       7    A
GPO              EQ     DIF4                                            GPO        5130    A
GPO                                                                     GPO        5131    A
GPO      *       ONE OPERAND MARKED, OR IN SCOPE OF MULTIPLY            GPO        5132    A
GPO                                                                     GPO        5133    A
GPO       DIF12  LX3    59-I.MP                                         GPO        5134    A
GPO              PL     X3,DIF4     IF ^M[LJ]                           GPO        5135    A
GPO              BX6    X4          OP = K                              GPO        5136    A
GPO              LX2    59-I.MP                                         FCC0640       8    A
GPO              MI     X2,DIF2     IF M[LK]      */ BOTH OPERANDS MARKED FCC0640       9    A
GPO              EQ     DIF4                                            GPO        5137    A
GPO                                                                     FCC0238AX    47    A
GPO       DIF13  SA3    "TB"+X4     LK = TB + K                         FCC0238AX    48    A
GPO              LX3    59-I.MP                                         FCC0238AX    49    A
GPO              MI     X3,DIF2     IF M[LK]                            FCC0238AX    50    A
GPO              SX6    X4          OP = K                              FCC0238AX    51    A
GPO              EQ     DIF4                                            FCC0238AX    52    A
GPO       TYIII  SPACE  2,10                                            GPO        5138    A
GPO      *       TYPE III INSTRUCTION ( LD, KLS, STT )                  GPO        5139    A
GPO                                                                     GPO        5140    A
GPO       DIF14  SA3    A5+B1       R2 = R1 + 1                         GPO        5141    A
GPO              LX3    -IH.RFP                                         GPO        5142    A
GPO              BX6    "RN"X3      OP = RF[R2]                         GPO        5143    A
GPO              SA2    "TB"+X6     LJ = TB + OP                        GPO        5144    A
GPO              ZR     X6,DIF2     IF RF[R2] = 0                       GPO        5145    A
GPO              LX2    59-I.MP                                         GPO        5146    A
GPO              UX4    B2,X7                                           GPO        5147    A
GPO              MI     X2,DIF2     IF M[LJ]                            GPO        5148    A
GPO              SX4    B2-OC.STT                                       GPO        5149    A
GPO              NZ     X4,DIF4     IF OC[R1W] " OC.STT                 GPO        5150    A
GPO              NZ     B6,DIF4     IF MULT " 0                         GPO        5151    A
GPO              SB4    OC.XMT      OC = OC.XMT   */ CHANGE STT TO AN XMT GPO      5152    A
GPO              EQ     DIF11                                           GPO        5153    A
GPO       DIF    SPACE  3,14                                            GPO        5154    A
GPO      **      PHASE 2, ADD INCREMENT EXPRESSION TO *IIT*             GPO        5155    A
GPO                                                                     GPO        5156    A
GPO       DIF15  MX7    0                                               GPO        5157    A
GPO              SX5    A7-B5       N = OI - O.RND  */ NUMBER OF INST   GPO        5158    A
GPO              SA7    A7+B1       OI = OI + 1;  [OI] = 0  */ TERMINATOR GPO      5159    A
GPO              LX5    2                                               GPO        5160    A
GPO              ALLOC  IIT,X5+2    ALLOC( IIT , 4*N+2 )                GPO        5161    A
GPO              SA3    O.TXT                                           GPO        5162    A
GPO              SA4    O.RND                                           GPO        5163    A
GPO              SX1    4           RN = 4                              GPO        5164    A
GPO              SA0    X2+B6       IB = O.IIT + OLD(L.IIT)             GPO        5165    A
GPO              SB4    B0          NLD = 0                             GPO        5166    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                SB5    X5+2        NWD = 4*N + 2                          GPO        5167    A
GPO                SB6    A0+4        II = IB + 4   */ STORE ADDRESS         GPO        5168    A
GPO                SA5    X4+1        R1 = [O.RND+1]                         GPO        5169    A
GPO                S"TB"  X3+3        TB = O.TXT + 3                         GPO        5170    A
GPO                                                                          GPO        5171    A
GPO       DIF16    BX7    "RN"X5      R = RI[R1]                             GPO        5172    A
GPO                SA2    DIFA                                               GPO        5173    A
GPO                SA3    "TB"+X7     LI = TB + R                            GPO        5174    A
GPO                UX7    B2,X5                                              GPO        5175    A
GPO                BX6    -X2*X3      M[LI] = 0                              GPO        5176    A
GPO                SA4    F.RDT+B2    DI = RDT(OC[R1])                       GPO        5177    A
GPO                BX7    X6+X1       HBP[LI] = RN  */ SAVE NEW R-NUMBER     GPO        5178    A
GPO                SA7    A3                                                 GPO        5179    A
GPO                BX6    X4          [II] = DI                              GPO        5180    A
GPO                SA6    B6                                                 GPO        5181    A
GPO                LX4    58-D.TYP                                           GPO        5182    A
GPO                MI     X4,DIF17    IF TYPE[DI] = III                      GPO        5183    A
GPO                LX4    1                                                  GPO        5184    A
GPO                MX7    0           R2W = 0                                GPO        5185    A
GPO                MI     X4,DIF18    IF TYPE[DI] = II                       GPO        5186    A
GPO                                                                          GPO        5187    A
GPO       *        ADJUST OPERAND R-NUMBERS OF TYPE I INSTRUCTION            GPO        5188    A
GPO                                                                          GPO        5189    A
GPO                LX5    -R1.RJP                                            GPO        5190    A
GPO                BX6    "RN"X5                                             GPO        5191    A
GPO                SA4    "TB"+X6                                            GPO        5192    A
GPO                LX5    R1.RJP-R1.RKP                                      GPO        5193    A
GPO                SX6    X4          RJ[R1] = HBP[TB+RJ[R1]]                GPO        5194    A
GPO                BX5    "RN"X5                                             GPO        5195    A
GPO                SA4    "TB"+X5                                            GPO        5196    A
GPO                LX6    R1.RJP                                             GPO        5197    A
GPO                SX3    X4          RK[R1] = HBP[TB+RK[R1]]                GPO        5198    A
GPO                LX3    R1.RKP                                             GPO        5199    A
GPO                PX6    B2,X6                                              GPO        5200    A
GPO                BX5    X6+X3                                              GPO        5201    A
GPO                EQ     DIF18                                              GPO        5202    A
GPO                                                                          GPO        5203    A
GPO       *        PROCESS TYPE III INSTRUCTION ( LD, KLS, STT )             GPO        5204    A
GPO                                                                          GPO        5205    A
GPO       DIF17    SA2    A3-2        R2 = LI - 2;  R2W = [R2]               GPO        5206    A
GPO                LX2    -IH.RFP                                            GPO        5207    A
GPO                BX7    "RN"X2                                             GPO        5208    A
GPO                SA4    "TB"+X7                                            GPO        5209    A
GPO                BX2    X0*X2                                              GPO        5210    A
GPO                SX7    X4          RF[R2W] = HBP[TB+RF[R2W]]              GPO        5211    A
GPO                LX6    59-D.LDP                                           GPO        5212    A
GPO                BX7    X2+X7                                              GPO        5213    A
GPO                LX7    IH.RFP                                             GPO        5214    A
GPO                PL     X6,DIF18    IF ^LD[II]                             GPO        5215    A
GPO                                                                          FCC0656      47    A
GPO                SA4    A3-B1       DIT = LI - 1                           FCC0656      48    A
GPO                SA2    DMASK                                              FCC0656      49    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO              BX6    -X2*X4      [II] = ^DMASK&[DIT]  */ PRESERVE FP,L2          FCC0656      50    A
GPO              SA6    B6                                                          FCC0656      51    A
GPO              LX3    59-I.INVP                                                   GPO        5216    A
GPO              SB4    B4+B1       NLD = NLD + 1                                   GPO        5217    A
GPO              MI     X3,DIF18    IF INV[LI]                                      GPO        5218    A
GPO                                                                                GPO        5219    A
GPO      *       SETUP FIRST HEADER WORD                                            GPO        5220    A
GPO                                                                                GPO        5221    A
GPO              LX5    -R1.INP                                                     GPO        5222    A
GPO              SA2    CAS                                                         GPO        5223    A
GPO              SX6    X5                                                          GPO        5224    A
GPO              LX2    IA.CAP      CA[IB] = CAS                                    GPO        5225    A
GPO              LX6    IA.UDIP     UDI[IB] = IN[R1]                                GPO        5226    A
GPO              PX3    B5,X6       NWD[IB] = NWD                                   GPO        5227    A
GPO              BX6    X2+X3                                                       GPO        5228    A
GPO              SB3    X1          LDI = RN                                        GPO        5229    A
GPO              LX5    R1.INP                                                      GPO        5230    A
GPO              SA6    A0                                                          GPO        5231    A
GPO                                                                                GPO        5232    A
GPO       DIF18  SA7    B6-B1       [II-1] = R2W                                    GPO        5233    A
GPO              BX5    X0*X5                                                       GPO        5234    A
GPO              MX6    0                                                           GPO        5235    A
GPO              BX7    X5+X1       RI[R1] = RN                                     GPO        5236    A
GPO              SA6    B6+B1       [II+1] = 0                                      GPO        5237    A
GPO              SA7    A7-B1       [II-2] = [R1]                                   GPO        5238    A
GPO              SX1    X1+4        RN = RN + 4                                     GPO        5239    A
GPO              SA5    A5+B1       R1 = R1 + 1                                     GPO        5240    A
GPO              SB6    B6+4                                                        GPO        5241    A
GPO              NZ     X5,DIF16    IF [RI] " 0                                     GPO        5242    A
GPO                                                                                GPO        5243    A
GPO      *       SETUP SECOND HEADER WORD                                           GPO        5244    A
GPO                                                                                GPO        5245    A
GPO              SX3    B3          ILD[IB+1] = LDI                                 GPO        5246    A
GPO              LX3    IB.ILDP                                                     GPO        5247    A
GPO              PX6    B4,X3       NLD[IB+1] = NLD                                 GPO        5248    A
GPO              SA6    A0+B1                                                       GPO        5249    A
GPO              EQ     DIF                                                         GPO        5250    A
GPO                                                                                GPO        5251    A
GPO       DIFA   BFMW   I,(M,HBP)                                                   GPO        5252    A
GPO       FIM    TITLE  FIM - FORM INCREMENT MODIFICATION LISTS                     GPO        5253    A
GPO      **      FIM - FORM INCREMENT MODIFICATION LISTS                            GPO        5254    A
GPO      *       SCAN *STC* ( *RD* VARS ) TO DEVELOP LIST OF *TP* INCREMENTS        GPO        5255    A
GPO      *       AT EACH RECURSIVE DEFINITION.                                      GPO        5256    A
GPO                                                                                GPO        5257    A
GPO              QUAL   FIM                                                         GPO        5258    A
GPO                                                                                GPO        5259    A
GPO       FIM    ROUTINE                                                            GPO        5260    A
GPO              S"TB"  X5+3        TB = O.SEQ + 3                                  GPO        5261    A
GPO              PX6    X4          NIS[BSW] = 0                                    GPO        5262    A
GPO              LX4    -BS.STCP                                                    GPO        5263    A
GPO              SA6    A4                                                          GPO        5264    A
GPO              SX7    X4          SC = STC[BSW]                                   GPO        5265    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                SA7    SC                                                    GPO        5266     A
GPO                                                                             GPO        5267     A
GPO      *         SEARCH *IIT* FOR FORMULAS DEPENDING ON THE VARIABLE IN THIS   GPO        5268     A
GPO      *         STORE.                                                        GPO        5269     A
GPO                                                                             GPO        5270     A
GPO       FIM1     SA1    O.IIT                                                 GPO        5271     A
GPO                SA2    L.IIT                                                 GPO        5272     A
GPO                SA4    "TB"+X7     LI = TB + SC                              GPO        5273     A
GPO                SB5    X1                                                    GPO        5274     A
GPO                LX4    -I.HBPP                                               GPO        5275     A
GPO                SA5    A4-3        R1 = LI - 3                               GPO        5276     A
GPO                BX7    X2          ITB = L.IIT                               GPO        5277     A
GPO                SA7    ITB                                                   GPO        5278     A
GPO                SA4    B5+X4       EV = IIT(HBP[LI])  */ INC EXPR VALUE      GPO        5279     A
GPO                BX6    X4                                                    GPO        5280     A
GPO                SA6    EV                                                    GPO        5281     A
GPO                LX5    -R1.INP                                               GPO        5282     A
GPO                SX7    X5          J = IN[R1]    */ *UDT* INDEX              GPO        5283     A
GPO                SA7    J                                                     GPO        5284     A
GPO                SA4    PIC         I = PIC                                   GPO        5285     A
GPO                                                                             GPO        5286     A
GPO       FIM2     SX6    X4+B1       PI = I + 1                                GPO        5287     A
GPO                SA4    B5+X4       PIW = IIT(I)                              GPO        5288     A
GPO                ZR     X4,FIM5     IF PIW = 0    */ END OF LIST              GPO        5289     A
GPO                SA1    A4+B1       PI = O.IIT + PI                           GPO        5290     A
GPO                LX1    -IA.UDIP                                              GPO        5291     A
GPO                BX7    X5-X1                                                 GPO        5292     A
GPO                SX3    X7+                                                   GPO        5293     A
GPO                LX4    -PI.LFP     I = LF[PIW]                               GPO        5294     A
GPO                NZ     X3,FIM2     IF UDI[PA] " J                            GPO        5295     A
GPO                                                                             GPO        5296     A
GPO      *         GET THE VALUE OF THE INCREMENT OF THIS *IP*                   GPO        5297     A
GPO                                                                             GPO        5298     A
GPO                SA6    PI                                                    GPO        5299     A
GPO                SX7    X4                                                    GPO        5300     A
GPO                SA7    A6+B1       PI(2) = I                                 GPO        5301     A
GPO                LX1    IA.UDIP                                               GPO        5302     A
GPO                RJ     EIE         EVALUATE  *DEL TP( EV ) *                 GPO        5303     A
GPO                                                                             GPO        5304     A
GPO                PRINT  FIM,(* -- INC VALUE = *O20),X7                         GPO        5305     A
GPO                                                                             GPO        5306     A
GPO                UX5    B2,X7                                                 GPO        5307     A
GPO                NZ     X5,FIM3     IF VAL[EW] " 0                            GPO        5308     A
GPO                                                                             GPO        5309     A
GPO      *         *IP* HAS A INC VALUE OF ZERO, CHANGE IT TO AN *INV* TEMP.     GPO        5310     A
GPO                                                                             GPO        5311     A
GPO                SB3    X6                                                    GPO        5312     A
GPO                SA1    O.TET                                                 GPO        5313     A
GPO                MX4    -T.IPFL                                               GPO        5314     A
GPO                SA2    X1+B3       TI = TET(CAS)                             GPO        5315     A
GPO                LX4    T.IPFP                                                GPO        5316     A
GPO                MX5    1                                                     GPO        5317     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                LX5    1+T.INVP                                                GPO        5318    A
GPO                BX2    X5+X2       INV[TI] = 1                                 GPO        5319    A
GPO                BX6    X4*X2       IPF[TI] = 0                                 GPO        5320    A
GPO                SA6    A2                                                      GPO        5321    A
GPO                SX7    B6                                                      GPO        5322    A
GPO                SA7    A3          L.IIT = OLD(L.IIT)                          GPO        5323    A
GPO                                                                               GPO        5324    A
GPO        FIM3    SA1    O.IIT                                                   GPO        5325    A
GPO                SA1    O.IIT                                                   GPO        5326    A
GPO                SA2    O.SEQ                                                   GPO        5327    A
GPO                SA4    PI+1                                                    GPO        5328    A
GPO                SA5    J                                                       GPO        5329    A
GPO                S"TB"  X2+3                                                    GPO        5330    A
GPO                SB5    X1                                                      GPO        5331    A
GPO                EQ     FIM2                                                    GPO        5332    A
GPO                                                                               GPO        5333    A
GPO        *       TERMINATE *TP* INC LIST, POINT FROM *ST* TO LIST.              GPO        5334    A
GPO                                                                               GPO        5335    A
GPO        FIM5    SA1    L.IIT                                                   GPO        5336    A
GPO                SA2    ITB                                                     GPO        5337    A
GPO                IX6    X1-X2                                                   GPO        5338    A
GPO                PX5    X2                                                      GPO        5339    A
GPO                AX7    B1,X6       NPI = (L.IIT-ITB) / 2                       GPO        5340    A
GPO                ZR     X6,FIM6     IF NPI = 0                                  GPO        5341    A
GPO                SB4    X7                                                      GPO        5342    A
GPO                SA4    BSW                                                     GPO        5343    A
GPO                PX5    B4,X5                                                   GPO        5344    A
GPO                UX6    B2,X4                                                   GPO        5345    A
GPO                SB3    B2+B4       NIS[BSW] = NIS[BSW] + NPI                   GPO        5346    A
GPO                PX7    B3,X4                                                   GPO        5347    A
GPO                SA7    A4                                                      GPO        5348    A
GPO                ADDWRD IIT,B0      ADDWRD( IIT , 0 ) */ LIST TERMINATOR        GPO        5349    A
GPO                SA4    O.SEQ                                                   GPO        5350    A
GPO                S"TB"  X4+3                                                    GPO        5351    A
GPO                                                                               GPO        5352    A
GPO        *       ADVANCE TO NEXT *ST* ON *STC*                                  GPO        5353    A
GPO                                                                               GPO        5354    A
GPO        FIM6    SA3    SC                                                      GPO        5355    A
GPO                SA2    "TB"+X3     LI = TB + SC                                GPO        5356    A
GPO                LX2    -I.STCP                                                 GPO        5357    A
GPO                SX7    X2          SC = STC[LI]                                GPO        5358    A
GPO                SA7    A3                                                      GPO        5359    A
GPO                SX2    X2                                                      GPO        5360    A
GPO                LX2    I.STCP                                                  GPO        5361    A
GPO                BX6    X5+X2       NPI[LI] = NPI;  HBP[LI] = ITB               GPO        5362    A
GPO                SA6    A2                                                      GPO        5363    A
GPO                NZ     X7,FIM1     IF SC " 0                                   GPO        5364    A
GPO                                                                               GPO        5365    A
GPO                EQ     FIM                                                     GPO        5366    A
GPO        EIE     TITLE  EIE - EVALUATE INCREMENT EXPRESSION                     GPO        5367    A
GPO        **      EIE - EVALUATE INCREMENT EXPRESSION OF A POLYNOMIAL            GPO        5368    A
GPO        *                                                                      GPO        5369    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO      *         ENTRY   (EV) = EXPRESSION VALUE ( 12/TYP,48/VAL )                GPO        5370     A
GPO      *                 (PI) = INDEX TO *IIT* OF INCREMENT FORMULA              GPO        5371     A
GPO      *                 (A1,X1) = PA, [PA] = FIRST WORD OF INCREMENT FORMULA    GPO        5372     A
GPO      *                                                                         GPO        5373     A
GPO      *         EXIT    *TP* INCREMENT VALUE TERM ADDED TO *IIT*                GPO        5374     A
GPO                                                                                GPO        5375     A
GPO       EIE0     BX5     X7                                                      GPO        5376     A
GPO                ALLOC   IIT,2       ALLOC( IIT , 2 );  I = OLD(L.IIT)           GPO        5377     A
GPO                SA4     CAS                                                     GPO        5378     A
GPO                BX7     X5                                                      GPO        5379     A
GPO                SX6     X4                                                      GPO        5380     A
GPO                SA6     X2+B6       IIT(I) = CAS  */ CA OF *IP*                 GPO        5381     A
GPO                SA7     A6+B1       IIT(I+1) = EW */ EXPR VALUE                 GPO        5382     A
GPO                                                                                GPO        5383     A
GPO       EIE      ROUTINE                                                         GPO        5384     A
GPO                SA2     A1+B1       PB = PA + 1                                 GPO        5385     A
GPO                SA3     EV                                                      GPO        5386     A
GPO                UX6     B2,X1       NW = NWD[PA]                                GPO        5387     A
GPO                UX4     B3,X2       NLD = NLD[PB]                               GPO        5388     A
GPO                LX6     -IA.CAP     CAS = CA[PA]                                GPO        5389     A
GPO                SB6     6                                                       GPO        5390     A
GPO                BX7     X3          EW = EV                                     GPO        5391     A
GPO                SA6     CAS                                                     GPO        5392     A
GPO                EQ      B2,B6,EIE0  IF NW = 6     */ MUST BE A *LD* ONLY        GPO        5393     A
GPO                                                                                GPO        5394     A
GPO                UX3     B4,X3       T = TYP[EV]                                 GPO        5395     A
GPO                NZ      B4,EIE1     IF T " 0      */ INCREMENT ^ A CON          GPO        5396     A
GPO                EQ      B3,B1,EIE9  IF NLD = 1    */ ONLY 1 *LD*                GPO        5397     A
GPO                                                                                GPO        5398     A
GPO      *         CHECK FOR *LD* / *XMTC* COMBINATION                             GPO        5399     A
GPO                                                                                GPO        5400     A
GPO       EIE1     GT      B3,B1,EIE2  IF NLD > 1                                  GPO        5401     A
GPO                SA4     A1+B6                                                   GPO        5402     A
GPO                UX6     B6,X4                                                   GPO        5403     A
GPO                SX5     B6-OC.XMTC                                              GPO        5404     A
GPO                NZ      X5,EIE2     IF OC[PA+6] " OC.XMTC                       GPO        5405     A
GPO                SB6     B2-10                                                   GPO        5406     A
GPO                NZ      B6,EIE2     IF NW " 10                                  GPO        5407     A
GPO                                                                                GPO        5408     A
GPO                SB5     B4-3        TYPE[EW] = 3 - T                            GPO        5409     A
GPO                SB4     -B5                                                     GPO        5410     A
GPO                PX7     B4,X7                                                   GPO        5411     A
GPO                EQ      EIE0                                                    GPO        5412     A
GPO                                                                                GPO        5413     A
GPO      *         COMPLEX CASE, SETUP *TST* = DEL TP(INC VAL) IN *TXT* AND        GPO        5414     A
GPO      *         CALL *SQZ* TO EVALUATE THE EXPRESSION.                          GPO        5415     A
GPO                                                                                GPO        5416     A
GPO       EIE2     ALLOC   TXT,10+B2   ALLOC( TXT , NW + 10 )                      GPO        5417          I
         -FCC0448
GPO       EIE2     ALLOC   TXT,14+B2   ALLOC( TXT , NW + 14 )                      FCC0448      62     A
GPO                SA1     EV                                                      GPO        5418     A
GPO                SB7     X2          TB = O.TXT;  TI = OLD(L.TXT)                GPO        5419     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                UX7    B4,X1        R2W = VAL[EV]                          GPO          5420     A
GPO                SB3    OC.XMT       OPC = OC.XMT                           GPO          5421     A
GPO                SX6    B1+B1        RI[R1] = 2                             GPO          5422     A
GPO                NZ     B4,EIE3      IF TYP[EV] " 0                         GPO          5423     A
GPO                                                                           GPO          5424     A
GPO                SB2    OC.S         OC = OC.S                              GPO          5425     A
GPO                LX7    R1.INP       IN[R1] = VAL[EV]                       GPO          5426     A
GPO                BX6    X7+X6                                               GPO          5427     A
GPO                SX7    0            R2W = 0                                GPO          5428     A
GPO                EQ     EIE4                                                GPO          5429     A
GPO                                                                           GPO          5430     A
GPO        EIE3    MX5    -IH.CAIHL                                           GPO          5431     A
GPO                BX4    X5*X7                                               GPO          5432     A
GPO                SB2    OC.LD                                               GPO          5433     A
GPO                BX7    -X5*X7       R2 = CAIH[EV]                          GPO          5434     A
GPO                LX4    R1.INP-IH.RFP                                       GPO          5435     A
GPO                BX6    X4+X6        R1 = TYII(0,RF[EV],0,2)                GPO          5436     A
GPO                EQ     B4,B1,EIE4   IF T = 1                               GPO          5437     A
GPO                SB3    OC.XMTC      OPC = OC.XMTC                          GPO          5438     A
GPO                                                                           GPO          5439     A
GPO       *        MOVE THE FORMULA TO *TXT*                                  GPO          5440     A
GPO                                                                           GPO          5441     A
GPO        EIE4    RJ     SRI          SRI( OC , R1 , R2 )                    GPO          5442          I
          -FCC0656
GPO        EIE4    RJ     SMI          SMI( OC , R1 , R2 )                    FCC0656        52     A
GPO                SA5    O.IIT                                               GPO          5443     A
GPO                SA2    PI                                                  GPO          5444     A
GPO                SB4    4                                                   GPO          5445     A
GPO                IX7    X5+X2                                               GPO          5446     A
GPO                SA1    X7           PA = O.IIT + PI                        GPO          5447     A
GPO                SB2    B1+B1                                               GPO          5448     A
GPO                UX6    B5,X1                                               GPO          5449     A
GPO                SB5    B5-B2        N = NWD[PA] - 2                        GPO          5450     A
GPO                SA5    A1+B2        PI = PA + 2                            GPO          5451     A
GPO                                                                           GPO          5452     A
GPO        EIE5    SA4    A5+B1                                               GPO          5453     A
GPO                SA3    A4+B1                                               GPO          5454     A
GPO                LX6    X5                                                  GPO          5455     A
GPO                BX1    "RN"X5       R = RI[R1]                             GPO          5456     A
GPO                LX7    X4                                                  GPO          5457     A
GPO                SB5    B5-B4        N = N - 4                              GPO          5458     A
GPO                SA6    B7+B6        TA = TB + TI; [TA] = [PI]              GPO          5459     A
GPO                SA7    A6+B1        [TA+1] = [PI+1]                        GPO          5460     A
GPO                BX6    X3                                                  GPO          5461     A
GPO                SA6    A7+B1        [TA+2] = [PA+2]                        GPO          5462     A
GPO                MX7    0                                                   GPO          5463     A
GPO                SA7    A6+B1        [TA+3] = 0                             GPO          5464     A
GPO                SB6    B6+B4        TI = TI + 4                            GPO          5465     A
GPO                SA5    A5+B4        PA = PA + 4                            GPO          5466     A
GPO                NZ     B5,EIE5      IF N " 0                               GPO          5467     A
GPO                                                                           GPO          5468     A
GPO                LX1    R1.RJP                                              FCC0448        63     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                SX5    B6                                                  FCC0448      64     A
GPO                SB2    OC.XMT                                              FCC0448      65     A
GPO                BX6    X1+X5                                               FCC0448      66     A
GPO                RJ     SRI         SRI( XMT  TI,R)                         FCC0448      67     A
GPO                SB2    OC.TST                                              GPO        5469     A
GPO                SA3    L.TET                                               GPO        5470     A
GPO                SX6    X1                                                  GPO        5471          I
          -FCC0448
GPO                SX6    B6-B4       R = TI - 4                              FCC0448      68     A
GPO                SA4    =XIT.                                               GPO        5472     A
GPO                LX3    IH.CAP                                              GPO        5473     A
GPO                BX7    X3+X4                                               GPO        5474     A
GPO                RJ     SRI         SRI( TST  R,L.TET,IT. )                 GPO        5475     A
GPO                SB2    B0                                                  GPO        5476     A
GPO                MX6    0                                                   GPO        5477     A
GPO                RJ     SRI         SRI( EOQ  0,0 )                         GPO        5478     A
GPO                                                                           GPO        5479     A
GPO       *        CHANGE *LD* OF THE *RD* VARIABLE TO AN *XMT/XMTC*          GPO        5480     A
GPO                                                                           GPO        5481     A
GPO                SA2    A1+B1       PB = PA + 1                             GPO        5482     A
GPO                LX2    -IB.ILDP                                            GPO        5483     A
GPO                SB6    X2+4        TI = ILD[PB] + 4                        GPO        5484     A
GPO                SX6    B1+B1                                               GPO        5485     A
GPO                SB2    B3                                                  GPO        5486     A
GPO                LX6    R1.RJP                                              GPO        5487     A
GPO                SX3    X2          RI = ILD[PB]                            GPO        5488     A
GPO                BX6    X6+X3                                               GPO        5489     A
GPO                RJ     SRI         SRI( OPC  2,0,RI )                      GPO        5490     A
GPO                                                                           GPO        5491     A
GPO                CALL   SIE         SQUEEZE INCREMENT EXPRESSION            GPO        5492          I
          -CCGNC
GPO                CALL   SIE#        SQUEEZE INCREMENT EXPRESSION            CCGNC       638     A
GPO                TRACE  EIE,RLIST                                           GPO        5493     A
GPO                SA4    L.TXT                                               GPO        5494     A
GPO                SX6    X4-5*4                                              GPO        5495          I
          -CCG0127
GPO                NZ     X6,EIE7     IF L.TXT " 20                           GPO        5496          I
          -CCG0127
GPO                SB3    X4-5*4                                              CCG0127      13     A
GPO                GT     B3,EIE7     IF MORE THAN 5 INSTRUCTIONS             CCG0127      14     A
GPO                SA1    O.TXT       T = O.TXT                               CCG0127      15     A
GPO                SA2    X1+4        R1L = [T+4]   */ R1 OF FIRST INSTRUCTION CCG0127     16     A
GPO                UX7    B2,X2                                               CCG0127      17     A
GPO                ZR     B3,EIE6Z    IF 5 INSTRUCTIONS                       CCG0127      18     A
GPO                                                                           CCG0127      19     A
GPO       *        4 INSTRUCTIONS, CHECK FOR CLR/S/FMA , TST                  CCG0127      20     A
GPO                                                                           CCG0127      21     A
GPO                SB4    B0          TYP = 0                                 CCG0127      22     A
GPO                NZ     X6,EIE6A                                            CCG0127      23     A
GPO                MX2    0                                                   CCG0127      24     A
GPO                EQ     EIE8                                                CCG0127      25     A
GPO                                                                           CCG0127      26     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO       EIE6A    SX6    B2-OC.S                                          CCG0127      27    A
GPO                NZ     X6,EIE7     IF NOT OC.S                          CCG0127      28    A
GPO                LX2    -R1.INP                                          CCG0127      29    A
GPO                MX3    42                                               CCG0127      30    A
GPO                BX2    -X3*X2                                           CCG0127      31    A
GPO                EQ     EIE8                                             CCG0127      32    A
GPO                                                                        GPO        5497    A
GPO      *         *TXT* SHOULD CONTAIN * BOS, LD, XMT/XMTC, TST, EOQ *    GPO        5498    A
GPO                                                                        GPO        5499    A
GPO                SA1    O.TXT       T = O.TXT                            GPO        5500          I
          -CCG0127
GPO                SA2    X1+4        R1L = [T+4]   */ R1 OF *LD*          GPO        5501          I
          -CCG0127
GPO                UX7    B2,X2                                            GPO        5502          I
          -CCG0127
GPO                SX6    B2-OC.LD                                         GPO        5503          I
          -CCG0127
GPO       EIE6Z    SX6    B2-OC.LD                                         CCG0127      33    A
GPO                AX7    R1.INP                                           GPO        5504    A
GPO                NZ     X6,EIE7     IF OC[R1L] " OC.LD                   GPO        5505          I
          -CCG0127
GPO                SA3    X1+8        R1 = [T+8]    */ R1 OF XMT/XMTC      GPO        5506          I
          -CCG0127
GPO                SB4    B1          TYP = 1                              CCG0127      34    A
GPO                ZR     X6,EIE6     IF OC[R1L] = OC.LD                   CCG0127      35    A
GPO                SB4    5                                                CCG0127      36    A
GPO                SX6    B2-OC.LDV                                        CCG0127      37    A
GPO                NZ     X6,EIE7     IF OC[R1L] .NE. OC.LDV               CCG0127      38    A
GPO                                                                        CCG0127      39    A
GPO       EIE6     SA3    X1+8        R1 = [T+8]  */ R1 OF XMT/XMTC        CCG0127      40    A
GPO                SA2    A2+B1       R2L = [T+5]   */ R2 WORD OF *LD*     GPO        5507    A
GPO                LX7    IH.RFP                                           GPO        5508    A
GPO                BX2    X7+X2       VAL = SHIFT(IN[R1],IH.RFP) + CAIH[R2L] GPO        5509    A
GPO                UX5    B2,X3                                            GPO        5510    A
GPO                SB3    OC.XMT                                           GPO        5511    A
GPO                PX7    B1,X2       EW = PACK(1,VAL)                     GPO        5512          I
          -CCG0127
GPO                EQ     B2,B3,EIE8  IF OC[R1] = OC.XMT                   GPO        5513    A
GPO                SB3    OC.XMTC                                          GPO        5514    A
GPO                SB4    B1+B1                                            GPO        5515          I
          -CCG0127
GPO                PX7    B4,X2       EW = PACK(2,VAL)                     GPO        5516          I
          -CCG0127
GPO                SB4    B4+B1       TYP = TYP + 1                        CCG0127      41    A
GPO                EQ     B2,B3,EIE8  IF OC[R1] = OC.XMTC                  GPO        5517    A
GPO                                                                        GPO        5518    A
GPO      *         SAVE EXPRESSION IN *MOD* FOR LATER MOVE TO HOLDING BLOCK. GPO      5519    A
GPO                                                                        GPO        5520    A
GPO       EIE7     ALLOC  MOD,X4-8    ALLOC( MOD , L.TXT - 8 )             GPO        5521    A
GPO                SA4    O.TXT                                            GPO        5522    A
GPO                SX3    X2+B6                                            GPO        5523    A
GPO                MOVE   X1,X4+4,X3  MOVE( L.TXT-8 , O.TXT+4 , O.MOD+OLD(L.MOD)) GPO   5524    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                MX1    1                                                    GPO        5525    A
GPO                SA5    IT.                                                  GPO        5526    A
GPO                LX1    1+T.INVP                                             GPO        5527    A
GPO                ADDWRD TET,X1      ADDWRD( TET , TETW(1,0,0,0) )             GPO        5528    A
GPO                SX3    X3-1                                                 GPO        5529    A
GPO                SB4    3                                                    GPO        5530        I
          -CCG0127
GPO                LX3    IH.CAP                                               GPO        5531        I
          -CCG0127
GPO                PX6    B4,X5                                                GPO        5532        I
          -CCG0127
GPO                BX7    X3+X6       EW = PACK( 3 , CAIH[L.TET-1,IT.] )       GPO        5533        I
          -CCG0127
GPO                                                                            GPO        5534        I
          -CCG0127
GPO        EIE8    SX6    4           L.TXT = 4                                GPO        5535        I
          -CCG0127
GPO                SB4    3           TYP = 3                                  CCG0127      42    A
GPO                LX3    IH.CAP                                               CCG0127      43    A
GPO                BX2    X3+X5       CAIH = CAIH[L.TET-1,IT.]                 CCG0127      44    A
GPO                                                                            CCG0127      45    A
GPO        EIE8    PX7    B4,X2       EW = PACK(TYP,CAIH)                      CCG0127      46    A
GPO                SX6    4           L.TXT = 4                                CCG0127      47    A
GPO                SA6    L.TXT                                                GPO        5536    A
GPO                EQ     EIE0                                                 GPO        5537    A
GPO        EIE     EJECT                                                       GPO        5538        I
          -CCG21
GPO        EIE     SPACE  4,20                                                 CCG21        29    A
GPO       *        EVALUATE EXPRESSION WITH CONSTANT INC VALUE AND 1 *LD*.     GPO        5539    A
GPO                                                                            GPO        5540    A
GPO        EIE9    SB7    A1+B1       TB = PA + 1                              GPO        5541    A
GPO                SA0    X3          LDV = VAL[EV] */ CON VALUE OF INC VARIABLE GPO        5542    A
GPO                MX1    D.TYL                                                GPO        5543    A
GPO                SB3    B1+B1                                                GPO        5544    A
GPO                SB4    B3+B3                                                GPO        5545    A
GPO                SA4    "TB"-B1     DI = TB - 1                              GPO        5546    A
GPO                SB6    B1+B1       I = 2                                    GPO        5547    A
GPO                                                                            GPO        5548    A
GPO       *        GET NEXT INSTRUCTION, JUMP TO PROCESSOR                     GPO        5549    A
GPO                                                                            GPO        5550    A
GPO        EIE10   SA4    A4+B4       DI = DI + 4                              GPO        5551    A
GPO                SA5    A4-B3       R1 = DI - 2                              GPO        5552    A
GPO                LX4    58-D.TYP                                             GPO        5553    A
GPO                BX6    X1*X4                                                GPO        5554    A
GPO                UX7    B5,X5                                                GPO        5555    A
GPO                NZ     X6,EIE11    IF TYPE[DI] " I                          GPO        5556    A
GPO                LX5    -R1.RJP                                              GPO        5557    A
GPO                BX6    "RN"X5                                               GPO        5558    A
GPO                LX5    R1.RJP-R1.RKP                                        GPO        5559    A
GPO                BX7    "RN"X5                                               GPO        5560    A
GPO                SA2    "TB"+X6     VJ = [TB+RJ[R1]]                         GPO        5561    A
GPO                SA3    "TB"+X7     VK = [TB+RK[R1]]                         GPO        5562    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                                                                                GPO        5563     A
GPO      EIE11    SA5    EIE.JT+B5                                                 GPO        5564     A
GPO               SB5    X5+                                                       GPO        5565     A
GPO               JP     B5          JUMP( EIE.JT( OC[R1] ) )                      GPO        5566     A
GPO                                                                                GPO        5567     A
GPO      *        STORE VALUE OF THIS INSTRUCTION                                  GPO        5568     A
GPO                                                                                GPO        5569     A
GPO      EIE12    SB6    B6+B4       I = I + 4                                     GPO        5570     A
GPO               SA7    A4+B1       LI = DI + 1;  [LI] = V                        GPO        5571     A
GPO               LT     B6,B2,EIE10 IF I < NW                                     GPO        5572     A
GPO                                                                                GPO        5573     A
GPO               MX1    -IH.CAL                                                   GPO        5574     A
GPO               SX7    X7          V = SETX(V)   */ ELIMINATE -0                 GPO        5575          I
         -CCG21
GPO               MX2    0                                                         CCG21        30     A
GPO               IX7    X2+X7       V = V + 0     */ ELIMINATE -0                 CCG21        31     A
GPO               BX3    X7                                                        CCG21        32     A
GPO               AX3    17                                                        CCG21        33     A
GPO               NZ     X3,EIE2     IF SHIFT(V,-17) " 0  */ INC VAL ^ A SMALL CON CCG21        34     A
GPO               BX6    -X1*X7                                                    GPO        5576     A
GPO               PX7    X6          EW = PACK(0,V)                                GPO        5577     A
GPO               EQ     EIE0                                                      GPO        5578     A
GPO                                                                                GPO        5579          I
         -CCG21
GPO      EIE      SPACE  3                                                         CCG21        35     A
GPO      *        INSTRUCTION PROCESSORS                                           GPO        5580     A
GPO                                                                                GPO        5581     A
GPO               PROCESS (LD,ILD)                                                 GPO        5582     A
GPO               SX7    A0          V = LDV                                       GPO        5583     A
GPO               EQ     EIE12                                                     GPO        5584     A
GPO                                                                                GPO        5585     A
GPO               PROCESS XMT                                                      GPO        5586          I
         -CCG41
GPO               PROCESS (XMT,SA)                                       002730    CCG41         1     A
GPO               BX7    X2          V = VJ                                        GPO        5587     A
GPO               EQ     EIE12                                                     GPO        5588     A
GPO                                                                                GPO        5589     A
GPO               PROCESS XMTC                                                     GPO        5590     A
GPO               BX7    -X2         V = -VJ                                       GPO        5591     A
GPO               EQ     EIE12                                                     GPO        5592     A
GPO                                                                                GPO        5593     A
GPO               PROCESS IA                                                       GPO        5594     A
GPO               IX7    X2+X3       V = VJ + VK                                   GPO        5595     A
GPO               EQ     EIE12                                                     GPO        5596     A
GPO                                                                                GPO        5597     A
GPO               PROCESS IS                                                       GPO        5598     A
GPO               IX7    X2-X3       V = VJ - VK                                   GPO        5599     A
GPO               EQ     EIE12                                                     GPO        5600     A
GPO                                                                                GPO        5601     A
GPO               PROCESS IM                                                       GPO        5602     A
GPO               IX7    X2*X3       V = VJ * VK                                   GPO        5603     A
GPO               EQ     EIE12                                                     GPO        5604     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                                                                    GPO          5605     A
GPO                PROCESS S                                           GPO          5606     A
GPO                LX7    -R1.INP                                      GPO          5607     A
GPO                SX7    X7          V = IN[R1]                       GPO          5608     A
GPO                EQ     EIE12                                        GPO          5609     A
GPO                                                                    GPO          5610     A
GPO                PROCESS FMA                                         GPO          5611     A
GPO                LX7    -R1.INP                                      GPO          5612     A
GPO                SB5    X7-1                                         GPO          5613     A
GPO                MX5    1                                            GPO          5614     A
GPO                AX7    B5,X5       V = SHIFT( 1S59 , -(IN[R1]-1) )  GPO          5615     A
GPO                EQ     EIE12                                        GPO          5616     A
GPO                                                                    GPO          5617     A
GPO                PROCESS SXT                                         GPO          5618     A
GPO                SA3    A4+3        R2 = DI + 3   */ ADVANCE TO *KLS* GPO          5619     A
GPO                SA4    A4+B4       DI = DI + 4                      GPO          5620     A
GPO                SB6    B6+B4       I = I + 4                        GPO          5621     A
GPO                LX3    -IH.CAP                                      GPO          5622     A
GPO                SB5    X3                                           GPO          5623     A
GPO                LX7    B5,X2       V = SHIFT( VJ , CA[R2] )         GPO          5624     A
GPO                EQ     EIE12                                        GPO          5625     A
GPO                                                                    FCC2271X       44     A
GPO                PROCESS UP                                          FCC2271X       45     A
GPO                BX7    X3          V = VK                           FCC2271X       46     A
GPO                EQ     EIE12                                        FCC2271X       47     A
GPO                                                                    GPO          5626     A
GPO       .STT     EQU    EIE2              */ CONTAINS AN *IH*        GPO          5627     A
GPO       .LDC     EQU    EIE2                                         CCG21          36     A
GPO       EIE.JT   SPACE  2                                             GPO          5628     A
GPO      *         COMBINED JUMP TABLE FOR *MIP* AND *EIE*             GPO          5629     A
GPO                                                                    GPO          5630     A
GPO                MACRO  OPR,NAM                                      GPO          5631     A
GPO                NOREF  /MIP/.NAM,.NAM                               GPO          5632     A
GPO                IF     DEF,/MIP/.NAM,2                              GPO          5633     A
GPO                EQ     /MIP/.NAM         .NAM                       GPO          5634     A
GPO                ELSE   1                                            GPO          5635     A
GPO                EQ     /MIP/MIP2                                    GPO          5636     A
GPO                IF     DEF,.NAM,2                                   GPO          5637     A
GPO      -         VFD    30/.NAM                                      GPO          5638     A
GPO                ELSE   1                                            GPO          5639     A
GPO      -         VFD    30/1S17+EIE                                  GPO          5640     A
GPO                ENDM                                                GPO          5641     A
GPO                                                                    GPO          5642     A
GPO       EIE.JT   BSS    0                                            GPO          5643     A
GPO      *CALL     OPRDEFS                                             GPO          5644     A
GPO                                                                    GPO          5645     A
GPO                QUAL                                                GPO          5646     A
GPO       MIP.JT   EQU    /FIM/EIE.JT                                  GPO          5647     A
GPO                                                                    GPO          5648     A
GPO                QUAL   CODE                                         GPO          5649     A
GPO                                                                    GPO          5650     A
GPO       FIM      EQU    /FIM/FIM                                     GPO          5651     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO       UPB      TITLE  UPB - UPDATE PROGRAM BLOCK TO REFLECT MODIFICATIONS          GPO        5652     A
GPO      **        UPB - UPDATE PROGRAM BLOCK TO REFLECT MODIFICATIONS FROM            GPO        5653     A
GPO      *         CODE MOTION AND STRENGTH REDUCTION PROCESSORS.                      GPO        5654     A
GPO      *         MERGE IN *LD*"S OF REMOVED INVARIANTS, POLYNOMIALS AND              GPO        5655     A
GPO      *         INCREMENT CODE.                                                     GPO        5656     A
GPO                                                                                    GPO        5657     A
GPO       UPB      ROUTINE                                                             GPO        5658     A
GPO                UX6    B2,X4       NPI = NIS[BSW]  */ N.POLY INCRS                  GPO        5659     A
GPO                MX3    -BS.NTETL                                                   GPO        5660     A
GPO                LX4    -BS.NTETP                                                   GPO        5661     A
GPO                BX7    -X3*X4      NTL = NTET[BSW]  */ N.TEMP LDS                   GPO        5662     A
GPO                SX1    B2+X7                                                       GPO        5663     A
GPO                SA7    NTL                                                         GPO        5664     A
GPO                SX6    B2+B2                                                       GPO        5665     A
GPO                IX5    X6+X7                                                       GPO        5666     A
GPO                LX5    3                                                           GPO        5667     A
GPO                ALLOC  MLT,X1      ALLOC( MLT , NTL + NPI )                        GPO        5668     A
GPO                ALLOC  MOD,X5      ALLOC( MOD , 8*NTL + 16*NPI )                   GPO        5669     A
GPO                SA3    O.MLT                                                       GPO        5670     A
GPO                SA5    O.SEQ                                                       GPO        5671     A
GPO                SB7    X2          MB = O.MOD; MB = OLD(L.MOD)                     GPO        5672          I
          -FCC0717
GPO                SA0    X3          MLI = O.MLT                                     GPO        5673          I
          -FCC0717
GPO                SA2    IT.                                                         GPO        5674          I
          -FCC0717
GPO                SA3    NTL                                                         GPO        5675          I
          -FCC0717
GPO                SB5    X5+2        SB = O.SEQ + 2                                  GPO        5676          I
          -FCC0717
GPO                SA1    TTI                                                         GPO        5677          I
          -FCC0717
GPO                ZR     X3,UPB4     IF NTL = 0                                      GPO        5678          I
          -FCC0717
GPO                                                                                    GPO        5679          I
          -FCC0717
GPO      *         SCAN *TET* AND SETUP *LD*"S OF REMOVED EXPRESSIONS                  GPO        5680          I
          -FCC0717
GPO                                                                                    GPO        5681          I
          -FCC0717
GPO                SA4    O.TET                                                       GPO        5682          I
          -FCC0717
GPO                SB2    X1                                                          GPO        5683          I
          -FCC0717
GPO                SA5    X4+B2       TI = O.TET + TTI  */ *TET* INDEX                GPO        5684          I
          -FCC0717
GPO                SX6    B2+X3       TTI = TTI + NTL  */ ADVANCE *TET* INDEX         GPO        5685          I
          -FCC0717
GPO                SA6    A1                                                          GPO        5686          I
          -FCC0717
GPO                SB4    X3          NT = NTL                                        GPO        5687          I
          -FCC0717

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                 SA1    UPBC                                                GPO        5688        I
          -FCC0717
GPO                                                                            GPO        5689        I
          -FCC0717
GPO        UPB1     LX5    -T.CAP                                              GPO        5690        I
          -FCC0717
GPO                 SX3    X5                                                  GPO        5691        I
          -FCC0717
GPO                 LX5    T.CAP-T.BIP                                         GPO        5692        I
          -FCC0717
GPO                 BX6    "RN"X5      R = BI[TI]    */ INSERT POINTER         GPO        5693        I
          -FCC0717
GPO                 LX3    IH.CAP                                              GPO        5694        I
          -FCC0717
GPO                 SA4    B5+X6       DI = SB + R                             GPO        5695        I
          -FCC0717
GPO                 SB2    OC.TLD                                              GPO        5696        I
          -FCC0717
GPO                 BX7    X2+X3       R2W = CAIH[CA[TI],IT.]                  GPO        5697        I
          -FCC0717
GPO                 BX3    X1*X4                                               GPO        5698        I
          -FCC0717
GPO                 NZ     X3,UPB2     IF STRS[DI] ! PI[DI] " 0                GPO        5699        I
          -FCC0717
GPO                                                                            GPO        5700        I
          -FCC0717
GPO       *         INSERT *LD* DIRECTLY IN *SEQ*                              GPO        5701        I
          -FCC0717
GPO                                                                            GPO        5702        I
          -FCC0717
GPO                 SA7    A4-B1       [DI-1] = R2W                            GPO        5703        I
          -FCC0717
GPO                 PX6    B2,X6                                               GPO        5704        I
          -FCC0717
GPO                 SA3    F.RDT+B2                                            GPO        5705        I
          -FCC0717
GPO                 BX7    X3                                                  GPO        5706        I
          -FCC0717
GPO                 SA6    A7-B1       [DI-2] = TYIII(OC.TLD,0,0,R)            GPO        5707        I
          -FCC0717
GPO                 SA7    A4          [DI] = RDT(OC.TLD)                      GPO        5708        I
          -FCC0717
GPO                 EQ     UPB3                                                GPO        5709        I
          -FCC0717
GPO                                                                            GPO        5710        I
          -FCC0717
GPO       *         SETUP *MOD* ENTRY FOR * LD / XMT *                         GPO        5711        I
          -FCC0717
GPO                                                                            GPO        5712        I
          -FCC0717
GPO        UPB2     SA3    UPBA                                                GPO        5713        I
          -FCC0717

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                 SX4    B6                                                    GPO        5714        I
          -FCC0717
GPO                 LX6    ML.IIP                                                GPO        5715        I
          -FCC0717
GPO                 LX4    ML.MTIP                                               GPO        5716        I
          -FCC0717
GPO                 IX4    X4+X6                                                 GPO        5717        I
          -FCC0717
GPO                 BX6    X3+X4                                                 GPO        5718        I
          -FCC0717
GPO                 SA6    A0          [MLI] = MCW(1,R,2,MI)                     GPO        5719        I
          -FCC0717
GPO                 SA0    A0+B1       MLI = MLI + 1                             GPO        5720        I
          -FCC0717
GPO                 SX6    B6                                                    GPO        5721        I
          -FCC0717
GPO                 RJ     SRI         SRI( TLD  R,R2W )                         GPO        5722        I
          -FCC0717
GPO                 SB2    OC.XMT                                                GPO        5723        I
          -FCC0717
GPO                 SX5    B6-4                                                  GPO        5724        I
          -FCC0717
GPO                 SX6    B6                                                    GPO        5725        I
          -FCC0717
GPO                 LX5    R1.RJP                                                GPO        5726        I
          -FCC0717
GPO                 BX6    X5+X6                                                 GPO        5727        I
          -FCC0717
GPO                 RJ     SRI         SRI( XMT  MI,MI-4 )                       GPO        5728        I
          -FCC0717
GPO                                                                              GPO        5729        I
          -FCC0717
GPO        UPB3     SB4    B4-B1       NT = NT - 1                               GPO        5730        I
          -FCC0717
GPO                 SA5    A5+B1       TI = TI + 1                               GPO        5731        I
          -FCC0717
GPO                 NZ     B4,UPB1     IF NT " 0                                 GPO        5732        I
          -FCC0717
GPO        UPB      SPACE  2,14                                                  GPO        5733        I
          -FCC0717
GPO       *         SCAN *STC* CHAIN AND SETUP INCREMENT CODE                    GPO        5734        I
          -FCC0717
GPO                                                                              GPO        5735        I
          -FCC0717
GPO        UPB4     SA4    BSW                                                   GPO        5736        I
          -FCC0717
GPO                 SB5    B5+B1       SB = SB + 1   */ O.SEQ + 3                GPO        5737        I
          -FCC0717
GPO                 UX6    B4,X4                                                 GPO        5738        I
          -FCC0717
GPO                 ZR     B4,UPB12    IF NIS[BSW] = 0  */ NO INCREMENTS         GPO        5739        I
          -FCC0717

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                 SA3    O.IIT                                                GPO        5740        I
          -FCC0717
GPO                 LX4    -BS.STCP                                             GPO        5741        I
          -FCC0717
GPO                 SX4    X4          SI = STC[BSW]                            GPO        5742        I
          -FCC0717
GPO                 SA1    X3          (A1) = [O.IIT]                           GPO        5743        I
          -FCC0717
GPO                                                                             GPO        5744        I
          -FCC0717
GPO        UPB5     SA3    B5+X4       LI = SB + SI                             GPO        5745        I
          -FCC0717
GPO                 UX6    B3,X3       NPI = NIS[LI]                            GPO        5746        I
          -FCC0717
GPO                 LX3    -I.HBPP                                              GPO        5747        I
          -FCC0717
GPO                 SB2    X3                                                   GPO        5748        I
          -FCC0717
GPO                 LX3    I.HBPP-I.STCP                                        GPO        5749        I
          -FCC0717
GPO                 SX7    X3          SC = STC[LI]  */ NEXT ON CHAIN           GPO        5750        I
          -FCC0717
GPO                 LX4    ML.IIP                                               GPO        5751        I
          -FCC0717
GPO                 SA7    SC                                                   GPO        5752        I
          -FCC0717
GPO                 ZR     B3,UPB11    IF NPI = 0    */ NO INCS AT THIS *ST*    GPO        5753        I
          -FCC0717
GPO                                                                             GPO        5754        I
          -FCC0717
GPO                 SA5    A1+B2       ILI = O.IIT + HBP[LI];  C = [ILI]  */ CA GPO        5755        I
          -FCC0717
GPO                 SX7    B6                                                   GPO        5756        I
          -FCC0717
GPO                 LX7    ML.MTIP                                              GPO        5757        I
          -FCC0717
GPO                 SB4    B6          MF = MI       */ INIT VALUE              GPO        5758        I
          -FCC0717
GPO                 BX6    X4+X7                                                GPO        5759        I
          -FCC0717
GPO                 SA6    A0          [MLI] = MCW(0,SI,0,MI)                   GPO        5760        I
          -FCC0717
GPO                 SA0    A0+B1       MLI = MLI + 1                            GPO        5761        I
          -FCC0717
GPO                                                                             GPO        5762        I
          -FCC0717
GPO       *         OUTPUT *TLD* OF *IP*                                        GPO        5763        I
          -FCC0717
GPO                                                                             GPO        5764        I
          -FCC0717
GPO        UPB6     SB2    OC.TLD                                               GPO        5765        I
          -FCC0717

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                 SX6    B6                                                    GPO        5766        I
          -FCC0717
GPO                 LX5    IH.CAP                                                GPO        5767        I
          -FCC0717
GPO                 BX7    X5+X2       R2W = CAIH[C,IT.]                         GPO        5768        I
          -FCC0717
GPO                 BX1    X7                                                    GPO        5769        I
          -FCC0717
GPO                 RJ     SRI         SRI( TLD  MI,R2W )                        GPO        5770        I
          -FCC0717
GPO                                                                              GPO        5771        I
          -FCC0717
GPO       *         OUTPUT STT / LD OF INCREMENT VALUE                           GPO        5772        I
          -FCC0717
GPO                                                                              GPO        5773        I
          -FCC0717
GPO                 SA5    A5+B1       ILI = ILI + 1                             GPO        5774        I
          -FCC0717
GPO                 UX7    B2,X5       T = TYP[ILI];  R2W = VAL[ILI]             GPO        5775        I
          -FCC0717
GPO                 SA3    UPBB+B2                                               GPO        5776        I
          -FCC0717
GPO                 SX6    B6                                                    GPO        5777        I
          -FCC0717
GPO                 NZ     B2,UPB7     IF T " 0                                  GPO        5778        I
          -FCC0717
GPO                                                                              GPO        5779        I
          -FCC0717
GPO                 LX7    IH.CAP                                                GPO        5780        I
          -FCC0717
GPO                 SX4    B6-4                                                  GPO        5781        I
          -FCC0717
GPO                 LX4    IH.RFP      C = VAL[ILI]  */ INC VALUE                GPO        5782        I
          -FCC0717
GPO                 BX7    X4+X7       R2W = IHW(0,MI-4,C,0)                     GPO        5783        I
          -FCC0717
GPO                 SB2    X3                                                    GPO        5784        I
          -FCC0717
GPO                 RJ     SII         SII( STT  MI,MI-4,C )                     GPO        5785        I
          -FCC0717
GPO                 EQ     UPB10                                                 GPO        5786        I
          -FCC0717
GPO                                                                              GPO        5787        I
          -FCC0717
GPO       *         TLD - REPLACE *CA* WITH EQUIVALENT                           GPO        5788        I
          -FCC0717
GPO                                                                              GPO        5789        I
          -FCC0717
GPO        UPB7     PL     X3,UPB8     IF T < 3                                  GPO        5790        I
          -FCC0717
GPO                 SA4    O.TET                                                 GPO        5791        I
          -FCC0717

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                 LX7    -IH.CAP                                                GPO        5792        I
            -FCC0717
GPO                 SB2    X7                                                     GPO        5793        I
            -FCC0717
GPO                 SA4    X4+B2                                                  GPO        5794        I
            -FCC0717
GPO                 LX4    -T.CAP                                                 GPO        5795        I
            -FCC0717
GPO                 SX7    X4          CA[R2W] = CA[TET(CA[R2W])                  GPO        5796        I
            -FCC0717
GPO                 LX7    IH.CAP                                                 GPO        5797        I
            -FCC0717
GPO                 BX7    X7+X2                                                  GPO        5798        I
            -FCC0717
GPO                 EQ     UPB9                                                   GPO        5799        I
            -FCC0717
GPO                                                                               GPO        5800        I
            -FCC0717
GPO         *       LD - SET *IN* FIELD IN R1 WORD                                GPO        5801        I
            -FCC0717
GPO                                                                               GPO        5802        I
            -FCC0717
GPO          UPB8   MX4    -IH.CAIHL                                              GPO        5803        I
            -FCC0717
GPO                 BX5    X4*X7       IH[R1] = RF[R2W]                           GPO        5804        I
            -FCC0717
GPO                 BX7    -X4*X7      R2W = CAIH[R2W]                            GPO        5805        I
            -FCC0717
GPO                 LX5    R1.INP-IH.RFP                                          GPO        5806        I
            -FCC0717
GPO                 BX6    X5+X6                                                  GPO        5807        I
            -FCC0717
GPO                                                                               GPO        5808        I
            -FCC0717
GPO          UPB9   SB2    X3          OC = SETX( UPBB(T) )                       GPO        5809        I
            -FCC0717
GPO                 AX3    18                                                     GPO        5810        I
            -FCC0717
GPO                 RJ     SRI         SRI( UPBB(T) , R1W,R2W )                   GPO        5811        I
            -FCC0656  -FCC0717
GPO                 RJ     SMI         SMI( UPBB(T) , R1W , R2W )                 FCC0656      53        I
            -FCC0717
GPO                                                                               GPO        5812        I
            -FCC0717
GPO         *       OUTPUT INCREMENT OF POLYNOMIAL ( TP = TP + INCV )             GPO        5813        I
            -FCC0717
GPO                                                                               GPO        5814        I
            -FCC0717
GPO                 SX5    B6-8                                                   GPO        5815        I
            -FCC0717
GPO                 SX4    B6-4                                                   GPO        5816        I
            -FCC0717

            0         1         2         3         4         5         6         7         8
            12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                 SX6    B6                                                    GPO        5817        I
            -FCC0717
GPO                 LX5    R1.RJP                                                GPO        5818        I
            -FCC0717
GPO                 SB2    X3          OC = SHIFT( UPBB(T) , -18 )               GPO        5819        I
            -FCC0717
GPO                 LX4    R1.RKP                                                GPO        5820        I
            -FCC0717
GPO                 BX3    X4+X5                                                 GPO        5821        I
            -FCC0717
GPO                 IX6    X3+X6                                                 GPO        5822        I
            -FCC0717
GPO                 RJ     SII         SII( OC  MI,MI-8,MI-4 )                   GPO        5823        I
            -FCC0717
GPO                                                                              GPO        5824        I
            -FCC0717
GPO         UPB10   SX6    B6-4                                                  GPO        5825        I
            -FCC0717
GPO                 SB2    OC.TST                                                GPO        5826        I
            -FCC0717
GPO                 BX7    X1                                                    GPO        5827        I
            -FCC0717
GPO                 RJ     SII         SII( TST  MI-4,R2W )                      GPO        5828        I
            -FCC0717
GPO                 SA5    A5+B1       ILI = ILI + 1                             GPO        5829        I
            -FCC0717
GPO                 NZ     X5,UPB6     IF [ILI] " 0                              GPO        5830        I
            -FCC0717
GPO                                                                              GPO        5831        I
            -FCC0717
GPO                 SA4    A0-B1                                                 GPO        5832        I
            -FCC0717
GPO                 SX3    B6-B4       NI = (MI-MF)/4  */ N.INSTRUCTIONS IN MOD  GPO        5833        I
            -FCC0717
GPO                 LX3    ML.NIP-2                                              GPO        5834        I
            -FCC0717
GPO                 BX6    X3+X4                                                 GPO        5835        I
            -FCC0717
GPO                 SA6    A4          NI[MIL-1] = NI                            GPO        5836        I
            -FCC0717
GPO                                                                              GPO        5837        I
            -FCC0717
GPO         UPB11   SA4    SC                                                    GPO        5838        I
            -FCC0717
GPO                 NZ     X4,UPB5     IF SC " 0                                 GPO        5839        I
            -FCC0717
GPO         UPB     SPACE  2,14                                                  GPO        5840        I
            -FCC0717
GPO         *       SET FINAL TABLE LENGTHS AND MERGE THE BLOCKS                 GPO        5841        I
            -FCC0717
GPO                                                                              GPO        5842        I
            -FCC0717

            0         1         2         3         4         5         6         7         8
            12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO       UPB12    SA1    O.MLT                                                GPO         5843          I
          -FCC0717
GPO                SB7    X2          MB = O.MOD;  MI = OLD(L.MOD)             FCC0717       61     A
GPO                SA2    IT.                                                  FCC0717       62     A
GPO                SA0    X3          MLI = O.MLT                              FCC0717       63     A
GPO                SB5    X5+3        SB = O.SEQ + 3                           FCC0717       64     A
GPO                                                                            FCC0717       65     A
GPO       *        SCAN *STC* AND SETUP INCREMENT CODE FOR THE *IP*"S.        FCC0717       66     A
GPO                                                                            FCC0717       67     A
GPO                SA4    BSW                                                  FCC0717       68     A
GPO                UX6    B4,X4                                                FCC0717       69     A
GPO                ZR     B4,UPB8     IF NIS[BSW] = 0  */ NO INCREMENTS        FCC0717       70     A
GPO                LX4    -BS.STCP                                             FCC0717       71     A
GPO                SA3    O.IIT                                                FCC0717       72     A
GPO                SX4    X4          SI = STC[BSW]                            FCC0717       73     A
GPO                SA1    X3          (A1) = [O.IIT]                           FCC0717       74     A
GPO                                                                            FCC0717       75     A
GPO       UPB1     SA3    B5+X4       LI = SB + SI                             FCC0717       76     A
GPO                UX6    B3,X3       NPI = NIS[LI]                            FCC0717       77     A
GPO                LX3    -I.HBPP                                              FCC0717       78     A
GPO                SB2    X3                                                   FCC0717       79     A
GPO                LX3    I.HBPP-I.STCP                                        FCC0717       80     A
GPO                SX7    X3          SC = STC[LI]  */ NEXT ON CHAIN           FCC0717       81     A
GPO                LX4    ML.IIP                                               FCC0717       82     A
GPO                SA7    SC                                                   FCC0717       83     A
GPO                ZR     B3,UPB7     IF NPI = 0    */ NO INCS AT THIS *ST*    FCC0717       84     A
GPO                                                                            FCC0717       85     A
GPO       *        LOOP THROUGH C LIST IN *IIT* AND OUTPUT INCREMENTS         FCC0717       86     A
GPO                                                                            FCC0717       87     A
GPO                SA5    A1+B2       ILI = O.IIT + HBP[LI];  C = [ILI]  */ CA FCC0717       88     A
GPO                SX7    B6                                                   FCC0717       89     A
GPO                LX7    ML.MTIP                                              FCC0717       90     A
GPO                SB4    B6          MF = MI       */ INIT VALUE              FCC0717       91     A
GPO                BX6    X4+X7                                                FCC0717       92     A
GPO                SA6    A0          [MLI] = MCW(0,SI,0,MI)                   FCC0717       93     A
GPO                SA0    A0+B1       MLI = MLI + 1                            FCC0717       94     A
GPO                                                                            FCC0717       95     A
GPO       *        OUTPUT *TLD* OF *IP*                                       FCC0717       96     A
GPO                                                                            FCC0717       97     A
GPO       UPB2     SB2    OC.TLD                                               FCC0717       98     A
GPO                SX6    B6                                                   FCC0717       99     A
GPO                LX5    IH.CAP                                               FCC0717      100     A
GPO                BX7    X5+X2       R2W = CAIH[C,IT.]                        FCC0717      101     A
GPO                BX1    X7          R2WS = R2W                               FCC0717      102     A
GPO                RJ     SRI         SRI( TLD  MI,R2W )                       FCC0717      103     A
GPO                                                                            FCC0717      104     A
GPO       *        OUTPUT STT / LD OF INCREMENT VALUE                         FCC0717      105     A
GPO                                                                            FCC0717      106     A
GPO                SA5    A5+B1       ILI = ILI + 1                            FCC0717      107     A
GPO                UX7    B2,X5       T = TYP[ILI];  R2W = VAL[ILI]            FCC0717      108     A
GPO                SA3    UPBB+B2                                              FCC0717      109     A
GPO                SX6    B6                                                   FCC0717      110     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                NZ     B2,UPB3     IF T " 0                                  FCC0717     111     A
GPO                SA4    O.TET                                                 CCG21        37     A
GPO                LX1    -IH.CAP                                               CCG21        38     A
GPO                SB2    X4                                                    CCG21        39     A
GPO                SA4    B2+X1       TW = TET(C)                               CCG21        40     A
GPO                LX1    IH.CAP                                                CCG21        41     A
GPO                LX4    59-T.RFP                                              CCG21        42     A
GPO                PL     X4,UPB2A    IF ^RF[TW]    */ NOT SHORT USE            CCG21        43     A
GPO                SA4    =XHO$LCM                                              CCG0212       5     A
GPO                PL     X4,UPB2C              */  IF  .NOT.  LCM=G            CCG0212       6     A
GPO                SA4    =XHO$OPT                                              CCG0212       7     A
GPO                MI     X4,UPB2A                */ IF OPT=2                   CCG0212       8     A
GPO                                                                             FCC0717     112     A
GPO                LX7    IH.CAP                                                FCC0717     113           I
          -CCG0212
GPO        UPB2C   LX7    IH.CAP                                                CCG0212       9     A
GPO                SX4    B6-4                                                  FCC0717     114     A
GPO                LX4    IH.RFP      C = VAL[ILI]  */ INC VALUE                FCC0717     115     A
GPO                BX7    X4+X7       R2W = IHW(0,MI-4,C,0)                     FCC0717     116     A
GPO                SB2    X3                                                    FCC0717     117     A
GPO                RJ     SII         SII( STT  MI,MI-4,C )                     FCC0717     118     A
GPO                EQ     UPB6                                                  FCC0717     119     A
GPO                                                                             CCG21        44     A
GPO        UPB2A   LX7    R1.INP                                                CCG21        45           I
          -CCG40
GPO        UPB2A   SX5    X7                                                    CCG40         5     A
GPO                PL     X5,UPB2B    IF C GE O                                 CCG40         6     A
GPO                BX7    -X5         C = ABS(C)                                CCG40         7     A
GPO                                                                             CCG40         8     A
GPO        UPB2B   LX7    R1.INP                                                CCG40         9     A
GPO                BX6    X7+X6                                                 CCG21        46     A
GPO                SB2    OC.S        OC = OC.S                                 CCG21        47     A
GPO                MX7    0                                                     CCG21        48     A
GPO                RJ     SRI         SRI( S  RI,C )                            CCG21        49     A
GPO                SX3    OC.IA                                                 CCG21        50     A
GPO                PL     X5,UPB5A    IF C WAS POSITIVE                         CCG40        10     A
GPO                SX3    OC.IS                                                 CCG40        11     A
GPO                EQ     UPB5A                                                 CCG21        51     A
GPO                                                                             FCC0717     120     A
GPO       *        TLD - REPLACE *CA* WITH EQUIVALENT                            FCC0717     121     A
GPO                                                                             FCC0717     122     A
GPO        UPB3    PL     X3,UPB4     IF T < 3                                  FCC0717     123           I
          -CCG0127
GPO        UPB3    PL     X3,UPB4     IF NOT *TLD*                              CCG0127      48     A
GPO                SA4    O.TET                                                 FCC0717     124     A
GPO                LX7    -IH.CAP                                               FCC0717     125     A
GPO                SB2    X7                                                    FCC0717     126     A
GPO                SA4    X4+B2                                                 FCC0717     127     A
GPO                LX4    -T.CAP                                                FCC0717     128     A
GPO                SX7    X4          CA[R2W] = CA[TET(CA[R2W])                 FCC0717     129     A
GPO                LX7    IH.CAP                                                FCC0717     130     A
GPO                BX7    X7+X2                                                 FCC0717     131     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                EQ     UPB5                                                  FCC0717     132     A
GPO                                                                             FCC0717     133     A
GPO      *         LD - SET *IN* FIELD IN R1 WORD                                FCC0717     134     A
GPO                                                                             FCC0717     135     A
GPO       UPB4     MX4    -IH.CAIHL                                              FCC0717     136     A
GPO                BX5    X4*X7        IH[R1] = RF[R2W]                          FCC0717     137           I
          -CCG0127
GPO                BX5    X4*X7        IN[R1] = RF[R2W]                          CCG0127      49     A
GPO                BX7    -X4*X7       R2W = CAIH[R2W]                           FCC0717     138     A
GPO                LX5    R1.INP-IH.RFP                                          FCC0717     139     A
GPO                BX6    X5+X6                                                  FCC0717     140     A
GPO                                                                             FCC0717     141     A
GPO       UPB5     SB2    X3           OC = SETX( UPBB(T) )                      FCC0717     142     A
GPO                AX3    18                                                     FCC0717     143     A
GPO                RJ     SMI          SMI( UPBB(T) , R1W , R2W )                FCC0717     144     A
GPO                                                                             FCC0717     145     A
GPO      *         OUTPUT INCREMENT OF POLYNOMIAL ( TP = TP + INCV )            FCC0717     146     A
GPO                                                                             FCC0717     147     A
GPO                SX5    B6-8                                                   FCC0717     148           I
          -CCG21
GPO       UPB5A    SX5    B6-8                                                   CCG21        52     A
GPO                SX4    B6-4                                                   FCC0717     149     A
GPO                SX6    B6                                                     FCC0717     150     A
GPO                LX5    R1.RJP                                                 FCC0717     151     A
GPO                SB2    X3           OC = SHIFT( UPBB(T) , -18 )               FCC0717     152     A
GPO                LX4    R1.RKP                                                 FCC0717     153     A
GPO                BX3    X4+X5                                                  FCC0717     154     A
GPO                IX6    X3+X6                                                  FCC0717     155     A
GPO                RJ     SII          SII( OC  MI,MI-8,MI-4 )                   FCC0717     156     A
GPO                                                                             FCC0717     157     A
GPO       UPB6     SX6    B6-4                                                   FCC0717     158     A
GPO                SB2    OC.TST                                                 FCC0717     159     A
GPO                BX7    X1           R2W = R2WS                                FCC0717     160     A
GPO                RJ     SII          SII( TST  MI-4,R2W )                      FCC0717     161     A
GPO                SA5    A5+B1        ILI = ILI + 1                             FCC0717     162     A
GPO                NZ     X5,UPB2      IF [ILI] " 0                              FCC0717     163     A
GPO                                                                             FCC0717     164     A
GPO                SA4    A0-B1                                                  FCC0717     165     A
GPO                SX3    B6-B4        NI = (MI-MF)/4  */ N.INSTRUCTIONS IN MOD  FCC0717     166     A
GPO                LX3    ML.NIP-2                                               FCC0717     167     A
GPO                BX6    X3+X4                                                  FCC0717     168     A
GPO                SA6    A4           NI[MIL-1] = NI                            FCC0717     169     A
GPO                                                                             FCC0717     170     A
GPO       UPB7     SA4    SC                                                     FCC0717     171     A
GPO                NZ     X4,UPB1      IF SC " 0                                 FCC0717     172     A
GPO       UPB      SPACE  2,10                                                   FCC0717     173     A
GPO      *         SCAN *TET* AND SETUP *TLD*"S OF REMOVED EXPRESSIONS           FCC0717     174     A
GPO                                                                             FCC0717     175     A
GPO       UPB8     SA3    NTL                                                    FCC0717     176     A
GPO                ZR     X3,UPB15     IF NTL = 0                                FCC0717     177     A
GPO                SA1    TTI                                                    FCC0717     178     A
GPO                SA4    O.TET                                                  FCC0717     179     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                SB2    X1                                                    FCC0717     180     A
GPO                SA5    X4+B2       TI = O.TET + TTI  */ *TET* INDEX          FCC0717     181     A
GPO                SX6    B2+X3       TTI = TTI + NTL  */ ADVANCE *TET* INDEX   FCC0717     182     A
GPO                SA6    A1                                                    FCC0717     183     A
GPO                SB4    X3          NT = NTL                                  FCC0717     184     A
GPO                SA1    UPBC                                                  FCC0717     185         I
          -CCG0214
GPO                                                                             FCC0717     186     A
GPO       UPB9     LX5    -T.CAP                                                FCC0717     187     A
GPO                SX3    X5                                                    FCC0717     188     A
GPO                LX5    T.CAP-T.BIP                                           FCC0717     189     A
GPO                BX6    "RN"X5      R = BI[TI]    */ INSERT POINTER           FCC0717     190     A
GPO                SA4    B5+X6       LI = SB + R                               FCC0717     191     A
GPO                LX3    IH.CAP                                                FCC0717     192     A
GPO                BX7    X2+X3       R2W = CAIH[CA[TI],IT.]                    FCC0717     193     A
GPO                LX4    59-I.AIPP                                             FCC0717     194     A
GPO                SA1    UPBC                                                  CCG0214       7     A
GPO                PL     X4,UPB11    IF ^AIP[LI]                               FCC0717     195     A
GPO                                                                             FCC0717     196     A
GPO       *        *TP* IS *AIP*, SEARCH FOR *TLD* OF INCREMENT IN *MOD* AND   FCC0717     197     A
GPO       *        FORM AN *XMT* FROM IT TO THE *TP*.  ( IO=I $ I=I+1 $ A(J,IO) )  FCC0717  198     A
GPO                                                                             FCC0717     199     A
GPO                SA3    B7+B1       I = 1                                     FCC0717     200         I
          -FCC2045
GPO       UPB10    SA3    A3+4        I = I + 4                                 FCC0717     201         I
          -FCC2045
GPO                SB3    A0-B1                                                 FCC2045       4     A
GPO                SB2    B7+B1                                                 FCC2045       5     A
GPO       UPB10    SA3    B3          MLW = [J]                                 FCC2045       6     A
GPO                SB3    B3-B1       J = J - 1     */ BACKWARDS SEARCH         FCC2045       7     A
GPO                LX3    -ML.IIP                                               FCC2045       8     A
GPO                BX4    "RN"X3                                                FCC2045       9     A
GPO                IX5    X6-X4                                                 FCC2045      10     A
GPO                MI     X5,UPB10    IF II[MLW] > R                            FCC2045      11     A
GPO                LX3    ML.IIP-ML.MTIP    I = MTI[MLW]                        FCC2045      12     A
GPO                SA3    B2+X3                                                 FCC2045      13         I
          -FCC2149
GPO                BX4    X3-X7                                                 FCC0717     202         I
          -FCC2149
GPO                NZ     X4,UPB10    IF MOD(I) " R2W                           FCC0717     203         I
          -FCC2149
GPO                                                                             FCC0717     204         I
          -FCC2149
GPO                SA3    A3-B1                                                 FCC0717     205         I
          -FCC2149
GPO                SA2    B2+X3                                        001900   FCC2149       1     A
GPO                LX3    ML.MTIP-ML.NIP    K = NI[MLW]                 001910   FCC2149       2     A
GPO                                                                    001920   FCC2149       3     A
GPO       UPB10A   BX4    X2-X7                                        001930   FCC2149       4     A
GPO                ZR     X4,UPB10B   IF MOD(I) = R2W                  001940   FCC2149       5     A
GPO                SX3    X3-1        K = K - 1                        001950   FCC2149       6     A
GPO                SA2    A2+4        I = I + 4                        001960   FCC2149       7     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                NZ     X3,UPB10A   IF K " 0                                  001970   FCC2149        8     A
GPO                EQ     UPB10                                                 001980   FCC2149        9          I
          -CCGA091
GPO                SA1    O.MLT                                                           CCGA091        6     A
GPO                BX1    -X1                                                             CCGA091        7     A
GPO                SX1    B3+X1                                                           CCGA091        8     A
GPO                PL     X1,UPB10                        */ IF MLT NOT EMPTY             CCGA091        9     A
GPO                EQ     UPB14                                                           CCGA091       10     A
GPO                                                                             001990   FCC2149       10     A
GPO       UPB10B   SA3    A2-B1                                                  002000   FCC2149       11     A
GPO                SX2    X7                RESTORE X2                           002010   FCC2149       12     A
GPO                BX5    "RN"X3      RJ = RI[MOD(I-1)]  */ RI OF TLD                     FCC0717      206     A
GPO                LX6    ML.IIP                                                          FCC0717      207     A
GPO                SX7    B6                                                              FCC0717      208     A
GPO                LX7    ML.MTIP                                                         FCC0717      209     A
GPO                BX6    X6+X7                                                           FCC0717      210     A
GPO                SX4    B1                                                              FCC0717      211     A
GPO                LX4    ML.NIP                                                          FCC0717      212     A
GPO                BX6    X4+X6                                                           FCC0717      213     A
GPO                MX4    1                                                               FCC0717      214     A
GPO                LX4    1+ML.DELP                                                       FCC0717      215     A
GPO                BX6    X4+X6                                                           FCC0717      216     A
GPO                SA6    A0          [MLI] = MCW(1,R,1,MI)                               FCC0717      217     A
GPO                SA0    A0+B1       MLI = MLI + 1                                       FCC0717      218     A
GPO                EQ     UPB13                                                           FCC0717      219     A
GPO                                                                                     FCC0717      220     A
GPO       UPB11    SA4    A4-B1       DI = LI - 1                                         FCC0717      221     A
GPO                SB2    OC.TLD                                                          FCC0717      222     A
GPO                BX3    X1*X4                                                           FCC0717      223     A
GPO                NZ     X3,UPB12    IF STRS[DI] ! PI[DI] " 0                            FCC0717      224     A
GPO                                                                                     FCC0717      225     A
GPO       *        INSERT *TLD* DIRECTLY IN *SEQ*                                        FCC0717      226     A
GPO                                                                                     FCC0717      227     A
GPO                SA7    A4-B1       [DI-1] = R2W                                        FCC0717      228     A
GPO                PX6    B2,X6                                                           FCC0717      229     A
GPO                SA3    F.RDT+B2                                                        FCC0717      230     A
GPO                BX7    X3                                                              FCC0717      231     A
GPO                SA6    A7-B1       [DI-2] = TYIII(OC.TLD,0,0,R)                        FCC0717      232     A
GPO                SA7    A4          [DI] = RDT(OC.TLD)                                  FCC0717      233     A
GPO                EQ     UPB14                                                           FCC0717      234     A
GPO                                                                                     FCC0717      235     A
GPO       *        SETUP *MOD* ENTRY FOR *TLD/XMT*                                       FCC0717      236     A
GPO                                                                                     FCC0717      237     A
GPO        UPB12   SA3    UPBA                                                            FCC0717      238     A
GPO                SX4    B6                                                              FCC0717      239     A
GPO                LX6    ML.IIP                                                          FCC0717      240     A
GPO                LX4    ML.MTIP                                                         FCC0717      241     A
GPO                IX4    X4+X6                                                           FCC0717      242     A
GPO                BX6    X3+X4                                                           FCC0717      243     A
GPO                SA6    A0          [MLI] = MCW(1,R,2,MI)                               FCC0717      244     A
GPO                SA0    A0+B1       MLI = MLI + 1                                       FCC0717      245     A
GPO                SX6    B6                                                              FCC0717      246     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                RJ     SRI         SRI( TLD  R,R2W )                    FCC0717      247     A
GPO                SX5    B6-4        RJ = MI-4                            FCC0717      248     A
GPO                                                                        FCC0717      249     A
GPO        UPB13   SX6    B6                                               FCC0717      250     A
GPO                LX5    R1.RJP                                           FCC0717      251     A
GPO                SB2    OC.XMT                                           FCC0717      252     A
GPO                MX7    0                                                FCC0717      253     A
GPO                BX6    X5+X6                                            FCC0717      254     A
GPO                RJ     SRI         SRI( XMT  MI,RJ )                    FCC0717      255     A
GPO                                                                        FCC0717      256     A
GPO        UPB14   SB4    B4-B1       NT = NT - 1                          FCC0717      257     A
GPO                SA5    A5+B1       TI = TI + 1                          FCC0717      258     A
GPO                NZ     B4,UPB9     IF NT " 0                            FCC0717      259     A
GPO        UPB     SPACE  2,10                                             FCC0717      260     A
GPO       *        SET FINAL TABLE LENGTHS AND MERGE THE BLOCKS            FCC0717      261     A
GPO                                                                        FCC0717      262     A
GPO        UPB15   SA1    O.MLT                                            FCC0717      263     A
GPO                SX2    A0                                               GPO         5844     A
GPO                SX6    B6                                               GPO         5845     A
GPO                SA6    L.MOD       L.MOD = MI                           GPO         5846     A
GPO                IX7    X2-X1                                            GPO         5847     A
GPO                MX6    1                                                GPO         5848     A
GPO                SA7    L.MLT       L.MLT = MLI - O.MOD                  GPO         5849     A
GPO                CALL   MPB         MERGE, SQUEEZE AND REWRITE BLOCK     GPO         5850     A
GPO                EQ     UPB                                              GPO         5851     A
GPO                                                                        GPO         5852     A
GPO        NTL     BSS    1           N.TET"S IN BLOCK                     GPO         5853     A
GPO        UPBA    BSS    0                                                GPO         5854     A
GPO                VFD    1/1                                              GPO         5855     A
GPO                POS    ML.NIP+2                                         GPO         5856     A
GPO                VFD    2/2,*P/0                                         GPO         5857     A
GPO                                                                        GPO         5858     A
GPO       *        UPBB - INCREMENT OPERATION TABLE                        GPO         5859     A
GPO                                                                        GPO         5860     A
GPO        INCOP   MACRO  A,B,C                                            GPO         5861     A
GPO                VFD    1/C,41/OC.B,18/OC.A                              GPO         5862     A
GPO                ENDM                                                    GPO         5863     A
GPO                                                                        GPO         5864     A
GPO        UPBB    BSS    0                                                GPO         5865     A
GPO                LOC    0                                                GPO         5866     A
GPO                INCOP  STT,S                                            GPO         5867     A
GPO                INCOP  LD,IA                                            GPO         5868     A
GPO                INCOP  LD,IS                                            GPO         5869     A
GPO                INCOP  TLD,IA,1                                         GPO         5870     A
GPO                INCOP  TLD,IS,1                                         GPO         5871     A
GPO                INCOP  LDV,IA                                           CCG0127       50     A
GPO                INCOP  LDV,IS                                           CCG0127       51     A
GPO                LOC    *O                                               GPO         5872     A
GPO                                                                        GPO         5873     A
GPO        UPBC    BFMW   D,(STRS,PI)   PS , PRS AND PI ( *DEF* MAY *PS* ) GPO         5874     A
GPO        GPO     TTL    GPO - GLOBAL PROGRAM OPTIMIZATION / SUBROUTINES GPO         5875     A
GPO        MPB     TITLE  MPB - MERGE PROGRAM BLOCK AND MOD LIST           GPO         5876     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO      **        MPB - MERGE PROGRAM BLOCK AND MOD LIST                          GPO        5877    A
GPO      *         MERGES A PROGRAM BLOCK DEFINED BY O.SEQ, L.SEQ AND THE MOD      GPO        5878    A
GPO      *         LIST ASSOCIATED WITH THE BLOCK, AND MOVES THE BLOCK TO *TXT*.   GPO        5879    A
GPO      *                                                                         GPO        5880    A
GPO      *         ENTRY  (X6) = 0 , MERGE BLOCK ONLY                              GPO        5881    A
GPO      *                (X6) > 0 , MERGE BLOCK AND CALL *SQZB* .                GPO        5882    A
GPO      *                (X6) < 0 , MERGE, CALL *SQZB* AND REWRITE TO *BLK*      GPO        5883    A
GPO      *                                                                         GPO        5884    A
GPO      *         EXIT   MERGED BLOCK IN *TXT*, R-NUMBERS IN CANONICAL ORDER.     GPO        5885    A
GPO      *                L.MLT = 0 , L.MOD = 4 AND L.TXT = 4 IF FLAG < 0 .       GPO        5886    A
GPO                                                                                GPO        5887    A
GPO       SB      MICRO  1,,/B6/     O.SEQ + 3                                     GPO        5888    A
GPO                                                                                GPO        5889    A
GPO       MPB0    SA1    MPBA                                                      GPO        5890    A
GPO               ZR     X1,MPB      IF MPBA = 0   */ NO SQZB CALL                 GPO        5891    A
GPO               MX6    0                                                         GPO        5892    A
GPO               CALL   SQZB                                                      GPO        5893          I
         -CCGNC
GPO               CALL   SQZB#                                                     CCGNC       639    A
GPO                                                                                GPO        5894    A
GPO               TRACE  MPB,RLIST                                                 GPO        5895    A
GPO                                                                                GPO        5896    A
GPO               SA1    MPBA                                                      GPO        5897    A
GPO               PL     X1,MPB      IF MPBA > 0                                   GPO        5898    A
GPO                                                                                GPO        5899    A
GPO               CALL   WTB         WRITE BLOCK BACK TO *BLK*                     GPO        5900    A
GPO                                                                                GPO        5901    A
GPO       MPB     ROUTINE                                                          GPO        5902    A
GPO               SA6    MPBA                                                      GPO        5903    A
GPO               TRACE  MPB,MLT                                                   GPO        5904    A
GPO               TRACE  MPB,RLIST,MOD                                             GPO        5905    A
GPO               SA5    L.MOD                                                     GPO        5906    A
GPO               SA4    L.SEQ                                                     GPO        5907    A
GPO               SX5    X5-8                                                      GPO        5908    A
GPO               IX1    X4+X5                                                     GPO        5909    A
GPO               ALLOC  TXT,X1      ALLOC( TXT , L.SEQ + L.MOD-8 )                GPO        5910    A
GPO                                                                                GPO        5911    A
GPO      *        INSERT MOD LIST INFO IN *SEQ*                                   GPO        5912    A
GPO                                                                                GPO        5913    A
GPO               SA4    O.SEQ                                                     GPO        5914    A
GPO               SA1    O.MOD                                                     GPO        5915    A
GPO               SX7    B6-4        PTL = OLD(L.TXT) - 4                          GPO        5916    A
GPO               SB3    3                                                         GPO        5917    A
GPO               IX6    X2+X7       TIA = O.TXT + PTL                             GPO        5918    A
GPO               SA6    TIA                                                       GPO        5919    A
GPO               S"TB"  X2+B3       TB = O.TXT + 3                                GPO        5920    A
GPO               MX0    1                                                         GPO        5921    A
GPO               SA3    O.MLT                                                     GPO        5922    A
GPO               S"SB"  X4+B3       SB = O.SEQ + 3                                GPO        5923    A
GPO               SA5    X3          I = [O.MLT]                                   GPO        5924    A
GPO               SA2    L.MLT                                                     GPO        5925    A
GPO               SB4    B0          MIA = 0       */ MIN INSERT ADDRESS           GPO        5926    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO      +        NZ     X7,*+1      IF PTL = 0  THEN MIA = *INFINITY*                GPO      5927     A
GPO               SB4    377777B     MIA = *INFINITY*  */ MIN INSERT ADDRESS          GPO      5928     A
GPO               LX1    ML.MTIP                                                      GPO      5929     A
GPO               ZR     X2,MPB2     IF L.MLT = 0                                     GPO      5930     A
GPO               SB5    X2+         N = L.MLT                                        GPO      5931     A
GPO       MPB     SPACE  1,10                                                         GPO      5932     A
GPO       MPB1    IX6    X5+X1       MTI[I] = MTI[I] + O.MOD  */ RELOCATE             GPO      5933     A
GPO               LX5    -ML.IIP                                                      GPO      5934     A
GPO               SA6    "SB"+X5     SI = SB + II[I];  [SI] = [I]  */ SAVE INSRT      GPO      5935     A
GPO               SB2    X5                                                           GPO      5936     A
GPO      +        SB5    B5-B1       N = N - 1                                        GPO      5937     A
GPO               GE     B2,B4,*+1   MIA = MIN( II[I] , MIA )                         GPO      5938     A
GPO               SB4    B2                                                           GPO      5939     A
GPO               SA4    A6-B3       R1 = SI - 3                                      GPO      5940     A
GPO               SA5    A5+B1       I = I + 1                                        GPO      5941     A
GPO               BX7    X0+X4       [R1] = MASK(1) ! [R1]  */ SET MOD FLAG           GPO      5942     A
GPO               SA7    A4                                                           GPO      5943     A
GPO               NZ     B5,MPB1     IF N " 0                                         GPO      5944     A
GPO       MPB     SPACE  2,10                                                         GPO      5945     A
GPO      *        MOVE INSTRUCTIONS FROM *SEQ* TO *TXT*, INSERT INSTRUCTIONS         GPO      5946     A
GPO      *        FROM *MOD*                                                         GPO      5947     A
GPO                                                                                  GPO      5948     A
GPO       MPB2    SA2    TIA                                                          GPO      5949     A
GPO               SA1    DMASK                                                        GPO      5950     A
GPO               SA0    X2          TI = TIA                                         GPO      5951     A
GPO               SX6    X2+B4       FIA = TIA + MIA                                  GPO      5952     A
GPO               SA6    FIA                                                          GPO      5953     A
GPO               SA5    "SB"-7      SI = SB-3 - 4                                    GPO      5954     A
GPO               SB4    4                                                            GPO      5955     A
GPO               EQ     MPB4                                                         GPO      5956     A
GPO                                                                                  GPO      5957     A
GPO      *        MOVE INSTRUCTION TO *TXT*                                          GPO      5958     A
GPO                                                                                  GPO      5959     A
GPO       MPB3    SA4    A5+B1       R2 = SI + 1                                      GPO      5960     A
GPO               BX6    X5                                                           GPO      5961     A
GPO               SA3    A4+B1       D = R2 + 1                                       GPO      5962     A
GPO               LX7    X4                                                           GPO      5963     A
GPO               SA6    A0          [TI] = [SI]                                      GPO      5964     A
GPO               UX5    B2,X5                                                        GPO      5965     A
GPO               SA7    A6+B1       [TI+1] = [R2]                                    GPO      5966     A
GPO               BX6    -X1*X3                                                       GPO      5967     A
GPO               SA6    A7+B1       [TI+2] = ^DMASK & [D]                            GPO      5968     A
GPO               MX7    0                                                            GPO      5969     A
GPO               SA7    A6+B1       [TI+3] = 0                                       GPO      5970     A
GPO               SX6    A7-"TB"     R = TI+3 - TB                                    GPO      5971     A
GPO               SA0    A0+B4       TI = TI + 4                                      GPO      5972     A
GPO               SA6    A5+B3       [SI+3] = R    */ NEW R-NUMBER                    GPO      5973     A
GPO               ZR     B2,MPB7     IF OC[SI] = 0 */ END OF SEQ                      GPO      5974     A
GPO                                                                                  GPO      5975     A
GPO       MPB4    SA5    A5+B4       SI = SI + 4                                      GPO      5976     A
GPO               PL     X5,MPB3     IF [SI] \ 0   */ NOT AN INSERT POINT             GPO      5977     A
GPO               SA2    A5+B3       LW = [SI+3]                                      GPO      5978     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO              BX6    -X0*X5                                                GPO        5979    A
GPO              LX2    59-ML.DELP                                            GPO        5980    A
GPO              MI     X2,MPB5     IF DEL[LW]    */ INST TO BE DELETED       GPO        5981    A
GPO                                                                           GPO        5982    A
GPO              SA4    A5+B1       R2 = SI + 1                               GPO        5983    A
GPO              SA3    A4+B1       D = R2 + 1                                GPO        5984    A
GPO              LX7    X4                                                    GPO        5985    A
GPO              SA6    A0          [TI] = [SI]                               GPO        5986    A
GPO              SA7    A6+B1       [TI+1] = [R2]                             GPO        5987    A
GPO              BX6    -X1*X3                                                GPO        5988    A
GPO              SA6    A7+B1       [TI+2] = ^DMASK & [D]                     GPO        5989    A
GPO              MX7    0                                                     GPO        5990    A
GPO              SA7    A6+B1       [TI+3] = 0                                GPO        5991    A
GPO              SX6    A7-"TB"     R = TI+3 - TB                             GPO        5992    A
GPO              SA0    A0+B4       TI = TI + 4                               GPO        5993    A
GPO              SA6    A5+B3       [SI+3] = R                                GPO        5994    A
GPO                                                                           GPO        5995    A
GPO       *      MOVE INSTRUCTIONS FROM *MOD* TO *TXT*                        GPO        5996    A
GPO                                                                           GPO        5997    A
GPO       MPB5   BX5    X2                                                    GPO        5998    A
GPO              LX2    1+ML.DELP-ML.MTIP                                     GPO        5999    A
GPO              SA4    X2          MI = MTI[LW]                              GPO        6000    A
GPO              LX2    ML.MTIP-ML.NIP                                        GPO        6001    A
GPO              SB2    X2          N = NI[LW]                                GPO        6002    A
GPO                                                                           GPO        6003    A
GPO       MPB6   SA3    A4+B1                                                 GPO        6004    A
GPO              BX6    -X4                                                   GPO        6005    A
GPO              SA2    A3+B1                                                 GPO        6006    A
GPO              SB2    B2-B1       N = N - 1                                 GPO        6007    A
GPO              SA6    A0          [TI] = -[MI]  */ FLIP R1 WORD             GPO        6008    A
GPO              BX7    X3                                                    GPO        6009    A
GPO              SA7    A6+B1       [TI+1] = [MI+1]                           GPO        6010    A
GPO              BX6    -X1*X2      [TI+2] = ^DMASK & [MI+2]                  GPO        6011    A
GPO              SA6    A7+B1                                                 GPO        6012    A
GPO              MX7    0                                                     GPO        6013    A
GPO              SA7    A6+B1       [TI+3] = 0                                GPO        6014    A
GPO              SX6    A7-"TB"     R = TI+3 - TB                             GPO        6015    A
GPO              SA6    A4+B3       [MI+3] = R    */ NEW R-NUMBER             GPO        6016    A
GPO              SA0    A0+B4       TI = TI + 4                               GPO        6017    A
GPO              LX2    -D.^DP+2                                              GPO        6018    A
GPO              SA4    A4+B4       MI = MI + 4                               GPO        6019    A
GPO              NZ     B2,MPB6     IF N " 0                                  GPO        6020    A
GPO                                                                           GPO        6021    A
GPO              PL     X5,MPB4     IF ^DEL[LW]   */ IF AN INSERT             GPO        6022    A
GPO              SX3    B4                                                    GPO        6023    A
GPO              BX2    X3*X2                                                 GPO        6024    A
GPO              IX6    X6-X2       R = R - 4*^D[MI-2]                        GPO        6025    A
GPO              SA6    A5+B3       [SI+3] = R    */ FOR *EXTERNAL* USE       GPO        6026    A
GPO              EQ     MPB4                                                  GPO        6027    A
GPO       MPB    SPACE  2,14                                                  GPO        6028    A
GPO       *      ADJUST R-NUMBERS TO BRING *TXT* TO CANONICAL FORM            GPO        6029    A
GPO                                                                           GPO        6030    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO       MPB7     SB5    A0+B3                                                GPO        6031    A
GPO                SX6    B5-"TB"     L.TXT = TI+3 - TB                        GPO        6032    A
GPO                SA5    O.MOD                                                GPO        6033    A
GPO                MX0    -R1.RIL+1                                            GPO        6034         I
          -CCG11
GPO                SX0    RN.MASK                                              CCG11       120    A
GPO                SA6    L.TXT                                                GPO        6035    A
GPO                LX0    1                                                    GPO        6036         I
          -CCG11
GPO                TRACE  MPB7,RLIST                                           GPO        6037    A
GPO                SA3    FIA                                                  GPO        6038    A
GPO                MX1    D.TYL                                                GPO        6039    A
GPO                SB2    B1+B1                                                GPO        6040    A
GPO                MI     X3,MPB0     IF FIA < 0    */ NO MODS TO BLOCK        GPO        6041    A
GPO                SA4    X3-2        DI = FIA - 2                             GPO        6042    A
GPO                SB5    X5+B3       MB = O.MOD + 3                           GPO        6043    A
GPO                MX6    0                                                    GPO        6044    A
GPO                SA0    "SB"-100000B      MSB = SB - 100000B */ MOD TO SEQ REFS GPO      6045    A
GPO                SA6    "SB"        [SB] = 0                                 GPO        6046    A
GPO                SA6    B5          [MB] = 0                                 GPO        6047    A
GPO                EQ     MPB10                                                GPO        6048    A
GPO                                                                            GPO        6049    A
GPO       *        ADJUST OPERAND R-NUMBERS OF INSTRUCTION FROM *SEQ*           GPO        6050    A
GPO                                                                            GPO        6051    A
GPO       MPB8     BX7    "RN"X5                                               GPO        6052    A
GPO                SA3    "SB"+X7     R = [SB+RI[R1]]                          GPO        6053    A
GPO                BX5    X0*X5                                                GPO        6054    A
GPO                SA2    A5+B1       R2 = R1 + 1                              GPO        6055    A
GPO                IX6    X5+X3       RI[R1] = R                               GPO        6056    A
GPO                NZ     X4,MPB9     IF TYPE[DI] " I                          GPO        6057    A
GPO                                                                            GPO        6058    A
GPO                LX6    -R1.RJP                                              GPO        6059    A
GPO                BX7    "RN"X6                                               GPO        6060    A
GPO                SA3    "SB"+X7     R = [SB+RJ[R1]]                          GPO        6061    A
GPO                BX6    X0*X6                                                GPO        6062    A
GPO                IX5    X6+X3       RJ[R1] = R                               GPO        6063    A
GPO                LX5    R1.RJP-R1.RKP                                        GPO        6064    A
GPO                BX7    "RN"X5                                               GPO        6065    A
GPO                SA3    "SB"+X7     R = [SB+RK[R1]]                          GPO        6066    A
GPO                BX5    X0*X5                                                GPO        6067    A
GPO                IX6    X5+X3       RK[R1] = R                               GPO        6068    A
GPO                SB0    0                                                    GPO        6069    A
GPO                LX6    R1.RKP                                               GPO        6070    A
GPO                                                                            GPO        6071    A
GPO       MPB9     LX2    -IH.RFP                                              GPO        6072    A
GPO                BX7    "RN"X2                                               GPO        6073    A
GPO                SA3    "SB"+X7     R = [SB+RF[R2]]                          GPO        6074    A
GPO                BX2    X0*X2                                                GPO        6075    A
GPO                SA6    A5                                                   GPO        6076    A
GPO                IX7    X2+X3       RF[R2] = R                               GPO        6077    A
GPO                LX7    IH.RFP                                               GPO        6078    A
GPO                SA7    A2                                                   GPO        6079    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                                                                    GPO     6080    A
GPO     *       ADVANCE TO NEXT INSTRUCTION                            GPO     6081    A
GPO                                                                    GPO     6082    A
GPO      MPB10  SA5    A4+B2       R1 = DI + 2;  RW = [R1]             GPO     6083    A
GPO             SA4    A5+B2       DI = R1 + 2                         GPO     6084    A
GPO             LX4    58-D.TYP                                        GPO     6085    A
GPO             BX6    -X1+X4                                          GPO     6086    A
GPO             BX4    X1*X4                                           GPO     6087    A
GPO             ZR     X6,MPB16    IF TYPE[DI] = IV                    GPO     6088    A
GPO             PL     X5,MPB8     IF RW > 0                           GPO     6089    A
GPO                                                                    GPO     6090    A
GPO     *       ADJUST OPERAND R-NUMBERS OF INSTRUCTION FROM *MOD*     GPO     6091    A
GPO                                                                    GPO     6092    A
GPO             BX5    -X5         RW = -RW                            GPO     6093    A
GPO             BX2    "RN"X5      R = RI[RW]                          GPO     6094    A
GPO             UX7    B3,X5                                           GPO     6095    A
GPO             SB7    B5                                              GPO     6096    A
GPO             LX7    -R1.RIL                                         GPO     6097    A
GPO             PL     X7,MPB11    B = IF R < 100000B THEN MB; ELSE MSB GPO     6098    A
GPO             SB7    A0                                              GPO     6099    A
GPO      MPB11  SA2    B7+X2       RI[RW] = [B+R]                      GPO     6100    A
GPO             BX6    X0*X5                                           GPO     6101    A
GPO             NZ     X4,MPB14    IF TYPE[DI] " I                     GPO     6102    A
GPO                                                                    GPO     6103    A
GPO             BX4    "RN"X7                                          GPO     6104    A
GPO             LX7    -R1.RIL                                         GPO     6105    A
GPO             SB7    B5                                              GPO     6106    A
GPO             PX6    B3,X2                                           GPO     6107    A
GPO             BX5    "RN"X7                                          GPO     6108    A
GPO             PL     X7,MPB12    B = IF R < 100000B THEN MB; ELSE MSB GPO     6109    A
GPO             SB7    A0                                              GPO     6110    A
GPO      MPB12  SA2    B7+X4       RK[RW] = [B+RK[RW]]                 GPO     6111    A
GPO             LX7    -R1.RIL                                         GPO     6112    A
GPO             SB7    B5                                              GPO     6113    A
GPO             LX2    R1.RKP                                          GPO     6114    A
GPO             BX6    X2+X6                                           GPO     6115    A
GPO             PL     X7,MPB13    B = IF R < 100000B THEN MB; ELSE MSB GPO     6116    A
GPO             SB7    A0                                              GPO     6117    A
GPO      MPB13  SA2    B7+X5       RJ[RW] = [B+RJ[RW]]                 GPO     6118    A
GPO             SB0    0                                               GPO     6119    A
GPO             LX2    R1.RJP                                          GPO     6120    A
GPO                                                                    GPO     6121    A
GPO      MPB14  SA3    A5+B1       R2 = R1 + 1                         GPO     6122    A
GPO             BX6    X2+X6                                           GPO     6123    A
GPO             LX3    -IH.RFP                                         GPO     6124    A
GPO             SA6    A5          [R1] = RW                           GPO     6125    A
GPO             BX7    "RN"X3                                          GPO     6126    A
GPO             SB4    X7                                              GPO     6127    A
GPO             LX7    -R1.RIL                                         GPO     6128    A
GPO             BX3    X0*X3                                           GPO     6129    A
GPO             SB7    B5                                              GPO     6130    A
GPO             PL     X7,MPB15    B = IF R < 100000B THEN MB; ELSE MSB GPO     6131    A

        0         1         2         3         4         5         6         7         8
        12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                SB7    A0                                                    GPO        6132    A
GPO       MPB15    SA2    B7+B4       RF[R2] = [B+RF[R2]]                       GPO        6133    A
GPO                IX6    X2+X3                                                 GPO        6134    A
GPO                LX6    IH.RFP                                                GPO        6135    A
GPO                SA6    A3                                                    GPO        6136    A
GPO                EQ     MPB10                                                 GPO        6137    A
GPO                                                                             GPO        6138    A
GPO       MPB16    PL     X5,MPB17    IF RW > 0                                 GPO        6139    A
GPO                BX7    -X5         [R1] = -RW                                GPO        6140    A
GPO                SA7    A5                                                    GPO        6141    A
GPO                                                                             GPO        6142    A
GPO       MPB17    UX6    B3,X5                                                 GPO        6143    A
GPO                NZ     B3,MPB10    IF OC[R1] " 0                             GPO        6144    A
GPO                                                                             GPO        6145    A
GPO                MX7    0                                                     GPO        6146    A
GPO                SX6    4                                                     GPO        6147    A
GPO                SA7    L.MLT       L.MLT = 0                                 GPO        6148    A
GPO                SA6    L.MOD       L.MOD = 4                                 GPO        6149    A
GPO                EQ     MPB0                                                  GPO        6150    A
GPO                                                                             GPO        6151    A
GPO       MPBA     BSS    1                                                     GPO        6152    A
GPO       FIA      BSS    1           FIRST INSERT ADDRESS                      GPO        6153    A
GPO       TIA      BSS    1           *TXT* INSERT ADDRESS ( O.TXT+OLD(L.TXT)   GPO        6154    A
GPO       MBT      SPACE  3,14        MBT                                       GPO        6155    A
GPO      **        MBT - MOVE BLOCK FROM *SEQ* TO *TXT*, CLEAR LINK WORDS.      GPO        6156    A
GPO                                                                             GPO        6157    A
GPO       MBT      ROUTINE                                                      GPO        6158    A
GPO                SA5    O.SEQ                                                 GPO        6159    A
GPO                SA2    X5+B1       R2 = [O.SEQ+1]                            GPO        6160    A
GPO                LX2    -R2.TXTLP                                             GPO        6161    A
GPO                ALLOC  TXT,X2-4    ALLOC( TXT , TXTL[R2]-4 )                 GPO        6162    A
GPO                SA1    O.SEQ                                                 GPO        6163    A
GPO                SB4    4                                                     GPO        6164    A
GPO                SA5    X1          R1 = [O.SEQ]                              GPO        6165    A
GPO                SB2    X3                                                    GPO        6166    A
GPO                SB7    X2          TI = [O.TXT]                              GPO        6167    A
GPO                                                                             GPO        6168    A
GPO       MBT2     SA4    A5+B1                                                 GPO        6169    A
GPO                SA3    A4+B1                                                 GPO        6170    A
GPO                BX6    X5                                                    GPO        6171    A
GPO                SA5    A5+B4                                                 GPO        6172    A
GPO                LX7    X4                                                    GPO        6173    A
GPO                SB2    B2-B4       I = I - 4                                 GPO        6174    A
GPO                SA6    B7                                                    GPO        6175    A
GPO                SB7    B7+B4                                                 GPO        6176    A
GPO                SA7    A6+B1                                                 GPO        6177    A
GPO                BX6    X3                                                    GPO        6178    A
GPO                SA6    A7+B1                                                 GPO        6179    A
GPO                MX7    0                                                     GPO        6180    A
GPO                SA7    A6+B1                                                 GPO        6181    A
GPO                NZ     B2,MBT2     IF I " 0                                  GPO        6182    A
GPO                                                                             GPO        6183    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

```
GPO                 EQ     MBT                                          GPO          6184     A
GPO                 QUAL                                                GPO          6185     A
GPO                                                                     GPO          6186     A
GPO                 ECHO   2,X=(EBV,FUD,MBT,MPB)                        GPO          6187          I
          -CCG23
GPO                 ECHO   2,X=(EBV,FUD,MBT,MPB,MPR)                    CCG23         125          I
          -CCG29
GPO                 ECHO   3,X=(EBV,FUD,MBT,MPB,MPR)                    CCG29          12     A
GPO                 IF     DEF,X,2                                      CCG29          13          I
          -CCG30
GPO                 IF     DEF,/CODE/_X,2                               CCG30           6     A
GPO                 ENTRY  X                                            GPO          6188          I
          -CCGNC
GPO        X        EQU    /CODE/X                                      GPO          6189          I
          -CCGNC
GPO                 ENTRY  ;A#                                          CCGNC         640     A
GPO       ;A#       EQU    /CODE/;A                                     CCGNC         641     A
GPO                                                                     GPO          6190     A
GPO                 END                                                 GPO          6191     A
```

SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - GPO

IDENTIFIER	TOTAL	ACTIVE
GPO	6191	5401
FCC0392	122	119
FCC0408	83	47
FCC0448	38	37
FCC0509	40	39
FCC0514	76	71
FCC0612	2	2
FCC0625	4	3
FCC0640	5	5
FCC0641	7	7
FCC0643	37	34
FCC0646	6	4
FCC0602	17	17
FCC0656	45	38
FCC0716	4	4
FCC0672	4	2
FCC0675	170	156
FCC0789	7	7
FCC0717	258	242
FCC0841	6	6
FCC0885	6	3
FCC0909	22	22
FCC1025	7	7
FCC0985	7	7
FCC1088	4	4
FCC1254	126	125
FCC1319	55	37
FCC1530X	12	12

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

RSM2763X	45	43
FCC1504X	23	23
FCC1532X	7	7
FCC1556X	13	7
CCG1	69	49
CCG2	12	12
CCG7	20	5
CCG9	3	2
FCC0885XA	11	11
FCC1584X	11	11
FCC1710X	22	22
CCGNC	121	118
CCG11	8	8
FCC1845X	2	2
FCC1858X	1	1
CCG16	5	3
FCC0238AX	47	47
CCG18	15	14
CCG21	46	45
CCG23	125	121
CCG28	14	14
FCC2045	10	9
CCG29	7	6
CCG30	5	5
CCG36	28	28
CCG34	4	4
FCCA462	2	2
CCG40	7	7
FCC1040A	5	5
FCC2149	12	11
CCG41	1	1
FCCA499X	51	51
FCCA506X	2	2
CCGA021	34	34
FCCA509X	16	16
FCCA513X	46	46
FCC2230X	7	7
FCC2271X	43	43
CCGA029	2	2
CCGA030	1	1
CCGA035	3	3
CCGA042	2	2
CCGA043	2	2
CCGA046	1	1
CCGA049	3	3
CCG0013	10	10
CCGA088	2	2
CCG0127	43	43
CCGA090	2	2
CCGA091	5	5
CCG0208	5	5
CCG0212	5	5

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GPO

                CCG0214                     1         1
                CCG0218                    15        15

                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

        GRA     *DECK     GRA                                                GRA          1     A
        GRA               IDENT  GRA                                         GRA          2     A
        GRA      GRA      TITLE  GRA - GLOBAL REGISTER ASSIGNMENT            GRA          3     A
        GRA     *CALL     SSTCALL                                            GRA          4     A
        GRA      B=GRA    RPVDEF                                             GRA          5     A
        GRA               PASS2TM                                            GRA          6           I
                -CCG1
        GRA                                                                  GRA          7           I
                -CCG1
        GRA     *IF       DEF,GRA,1                                          GRA          8     A
        GRA               TRACER (UDT,MTA,MFA,CLB,SUP,SXC)                   GRA          9           I
                -FCC1039
        GRA               TRACER (UDT,SEE,MTA,DXA,MFA,CLB,SUP,SXC)           FCC1039      4     A
        GRA      GRA      SPACE  2                                           GRA         10     A
        GRA     **        GRA - GLOBAL REGISTER ASSIGNMENT                   GRA         11     A
        GRA     *                                                            GRA         12     A
        GRA     *         S. I. JASIK - CDC - SUNNYVALE - SPRING 74          GRA         13     A
        GRA     *                                                            GRA         14     A
        GRA     *         *GRA* IS CALLED BY *GPO* TO ASSIGN REGISTERS ACCROSS THE  GRA  15     A
        GRA     *         BODY OF A LOOP AND SETUP APPROPIATE INITIALIZATION CODE.  GRA  16     A
        GRA     *                                                            GRA         17     A
        GRA     *         OPTIMIZATIONS PERFORMED BY *GRA* INCLUDE -         GRA         18     A
        GRA     *         LOAD/STORE MOTION OF SCALAR VARIABLES AND CONSTANTS AND THEIR  GRA  19     A
        GRA     *         ASSIGNMENT TO REGISTERS.                           GRA         20     A
        GRA     *         PREFETCHING OF INDEXED LOADS ON THE CRITICAL PATH IN SMALL  GRA  21     A
        GRA     *         INNERMOST LOOPS.                                   GRA         22     A
        GRA     *         CODE SIZE REDUCTION IN THE LOOP BODY BY ASSIGNING *SCM*  GRA  23     A
        GRA     *         ADDRESS"S AND CONSTANTS TO B-REGISTERS.            GRA         24     A
        GRA     *         INCREMENT REDUCTION OF *IP*"S BY ASSIGNING THE DIFFERENCE OF  GRA  25     A
        GRA     *         TWO SIMILAR INTEGER POLYNOMIALS TO A B-REGISTER.   GRA         26     A
        GRA     *         TEST REPLACEMENT OF THE LOOP CONTROL VARIABLE WITH A LINEAR  GRA  27     A
        GRA     *         FUNCTION OF IT WHEN POSSIBLE.                      GRA         28     A
        GRA     *         REMOVAL OF INCREMENT VARIABLES WHICH HAVE NO USES. GRA         29     A
        GRA                                                                  GRA         30     A
        GRA     **        DEBUGGING SNAP LABELS -                            GRA         31     A
        GRA     *                                                            GRA         32     A
        GRA     *         TRACER (UDT,ERC,SEE,MTA,DXA,DAA,MFA,CLB,SUP,SXC)   GRA         33     A
        GRA     *                                                            GRA         34     A
        GRA     *         BASIC SNAPS - (UDT,MFA,CLB,SUP,SXC)                GRA         35           I
                -FCC1039
        GRA     *         BASIC SNAPS - UDT,SEE,MTA,DXA,DAA,CLB,SUP,SXC      FCC1039      5     A
        GRA                                                                  GRA         36     A
        GRA               LIST   -R                                          GRA         37     A
        GRA                                                                  GRA         38           I
                -CCGNC
        GRA               TABLES BIT,BST,IIT,SEQ,TET,TXT                     GRA         39           I
                -CCG1     -CCGNC

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                TABLES MOD,PIT,PSI,RCT,RND,RXI                                GRA          40        I
          -CCG1    -CCGNC
GRA                ECHO   1,Z=(BIT,BST,IIT,MOD,PIT,PSI,RCT,RND,RXI,SEQ,TET,TXT)   CCG1         89        I
          -CCGNC
GRA                EXT    O.Z,L.Z                                                CCG1         90        I
          -CCGNC
GRA                                                                              GRA          41        I
          -CCGNC
GRA                EXT    BSI,BVL,DMASK,F.RDT,NPB,SRI                            GRA          42        I
          -CCGNC
GRA                                                                              CCGNC       642    A
GRA                TABLES (BIT,BST,IIT,MOD,PIT,PSI,RCT,RND,RXI,SEQ,TET,TXT)     CCGNC       643    A
GRA                                                                              CCGNC       644    A
GRA                EXT#   (BSI,BSW,BVL,DMASK,L.HB,N.GT,HBI,PC,SRI)               CCGNC       645    A
GRA                                                                              GRA          43    A
GRA       MAX.INC  EQU    101B        MAX CONSTANT INCREMENT VALUE FOR PREFETCHS GRA          44    A
GRA      *                          OF ARRAYS IN LABELED COMMON OR LOCAL STORAGE. GRA        45    A
GRA                                                                              GRA          46    A
GRA       BA.BIP   EQU    36                                                     GRA          47    A
GRA       BA.LENP  EQU    18                                                     GRA          48    A
GRA       BA.BIL   EQU    18                                                     FCCA509X     24    A
GRA       BA.LENL  EQU    18                                                     FCCA509X     25    A
GRA                                                                              GRA          49    A
GRA      *CALL     UDTBLD                                                        GRA          50    A
GRA      *CALL     GPOCOM                                                        GRA          51    A
GRA       FLAGS    TITLE  LOCAL VARIABLES                                       GRA          52    A
GRA                USE    /TABLES/                                              GRA          53        I
          -CCG1
GRA                USE    /CCG.SCR/                                             CCG1         91        I
          -CCG18
GRA                USE    /CCGSCR/                                              CCG18       100    A
GRA       SCR      BSS    100B        *IA* LIST ( DUM - MTA ), CON LIST  ( MFA )  GRA          54    A
GRA      *                           AND SCRATCH FOR *BDT/MCG*                  GRA          55    A
GRA                                                                              GRA          56    A
GRA       F.CLC    BSS    0          ?FWA OF CLEARED CELLS                       GRA          57    A
GRA                                                                              GRA          58    A
GRA       RAT      BSS    24          REGISTER ASSIGNMENT TABLE ( INITIAL VALUES GRA          59    A
GRA       RVT      BSS    24          REGISTER VALUE TABLE ( R-NUMBER IN REG )   GRA          60    A
GRA                                                                              GRA          61    A
GRA      *                       ERC                                             GRA          62    A
GRA       IAI      BSS    1           *IA* LIST INDEX                            GRA          63    A
GRA       TRA      BSS    1           TEMP/RJRS REG ASSIGNMENTS ( BIT MASK )     GRA          64    A
GRA       APF      BSS    1           APLN BITS OF BLOCK                         GRA          65    A
GRA       FBA      BSS    1           "0 IF FINAL B-ASSIGNMENTS FROM INNER LOOP  GRA          66    A
GRA       LCT      BSS    1           L.RCT AT END OF *ERC*                      GRA          67    A
GRA      *                       SEE                                             GRA          68    A
GRA       NVC      BSS    1           N. VALUE CANIDATES                         GRA          69    A
GRA       TUD      BSS    1           SUSE[CI]-STC[UI] FOR LOOP CV               GRA          70    A
GRA      *                       MTA                                             GRA          71    A
GRA       ABR      BSS    1           SET OF AVAIL B-REGS ( PACK(0,377B)         GRA          72    A
GRA       NAB      BSS    1           N. AVAILABLE B-REGS                        GRA          73    A
GRA       NTA      BSS    1           N. TENATIVE B-ASSIGNMENTS                  GRA          74    A

          0        1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA      TRD      BSS    1           TEST REPLACEMENT DECISION                    GRA        75    A
GRA     *                        DXA                                             GRA        76    A
GRA      NXC      BSS    1           N. X-CANIDATES                               GRA        77    A
GRA      XCT      BSS    5           X-REG CANIDATE TABLE                         GRA        78    A
GRA      MLW      BSS    1           MAX REG WIDTH AFTER X-ASSIGNMENT             GRA        79    A
GRA     *                        CMR                                             GRA        80    A
GRA      N.LD     BSS    2           N. REMAINING LOADS AFTER REGISTER ASSIGNMEN  GRA        81    A
GRA      N.ST     EQU    N.LD+1      MAX NUMBER OF ST"S IN A BLOCK                GRA        82    A
GRA     *                        DAA                                             GRA        83    A
GRA      MAA      BSS    1           MAX NUMBER OF A-ASSIGNMENTS                  GRA        84    A
GRA      NAA      BSS    1           N. A-ASSIGNMENTS                             GRA        85    A
GRA      ACT      BSS    8           A-CANIDATE TABLE                             GRA        86    A
GRA     *                            24/,18/INC PTR,18/CANIDATE INDEX ( *RCT* )   GRA        87    A
GRA     *                        MFA                                             GRA        88    A
GRA      CL       BSS    1           CON LIST POINTER                             GRA        89    A
GRA      NAC      BSS    1           N. ADDRESS CANIDATES                         GRA        90    A
GRA      ATI      BSS    1           INDEX TO *AI* TABLE ( LWA+1 )                GRA        91    A
GRA      CAIH     BSS    1           CA,IH OF BASE REGISTER                       GRA        92    A
GRA      PMF      BSS    1           PARTIAL MODIFICATION FLAG ( LD REG+CON "S )  GRA        93    A
GRA      CIT      BSS    5           IH CLASS INDEX TABLE                         GRA        94    A
GRA                                                                              GRA        95    A
GRA      L.CLC    EQU    *-F.CLC    'LWA+1 OF CLEARED CELLS                       GRA        96    A
GRA                                                                              GRA        97    A
GRA     *                        SUP                                             GRA        98    A
GRA      N        BSS    1           N. WDS IN POLY FORMULA                       GRA        99    A
GRA      RR       EQU    N           R-NUMBER OF LAST RESULT                      GRA       100    A
GRA     *                        CLB                                             GRA       101    A
GRA      NXS      BSS    1           N. EXTRA WORDS NEEDED                        GRA       102    A
GRA      MRU      BSS    1           MACHINE REGS USED BY *MCG*                   GRA       103    A
GRA      NPS      BSS    1           N. POST STORES                               GRA       104    A
GRA      PST      BSS    7+4         POST STORE LIST , 24/,18/REGNO,18/*RCT* ORD  GRA       105    A
GRA     *                        IRA                                             GRA       106    A
GRA      LBJ      BSS    1           "0 IF SEQUENCE CONTAINS THE LOOP BACK JUMP   GRA       107         I
        -FCC0985
GRA      LBN      BSS    1           R1 WORD OF LAST ENCOUNTERED BOS/EOS          FCC0985    14    A
GRA      DTL      BSS    1           DEF TEXT LENGTH ( BOS + DEFS + EOQ )         FCC2051     5    A
GRA      LBJF     BSS    1           LOOP BACK JUMP ENCOUNTERED                   CCG0119     6    A
GRA               USE    0                                                        GRA       108    A
GRA      RCT      TITLE  RCT - REGISTER CANIDATE TABLE FORMAT                     GRA       109    A
GRA     **        RCT - REGISTER CANIDATE TABLE FORMAT                            GRA       110    A
GRA     *                                                                        GRA       111    A
GRA     *         THE *RCT* IS A DICTIONARY OF ALL EXPRESSIONS ( MEMORY          GRA       112    A
GRA     *         REFERENCES, CONSTANTS AND ADDRESS"S ) THAT ARE CANIDATES FOR   GRA       113    A
GRA     *         REGISTER ASSIGNMENT. EACH ENTRY CONSISTS OF 3 WORDS.           GRA       114    A
GRA      CA       SPACE  2                                                        GRA       115    A
GRA     **        CA. - FIRST WORD OF CANIDATE TABLE                              GRA       116    A
GRA                                                                              GRA       117    A
GRA               DESCRIBE CA.,60                                                GRA       118    A
GRA      RA       DEFINE 1           REGISTER ASSIGNED OR ADDRESS MODIFIED        GRA       119    A
GRA      IRA      DEFINE 1           INITIAL REGISTER ASSIGNMENT ( *ERC/RS* )     GRA       120    A
GRA                                                                              GRA       121    A
GRA     *         CANIDATE TYPE                                                  GRA       122    A

        0         1         2         3         4         5         6         7         8
        12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                                                                                GRA          123     A
GRA       VC       DEFINE 1          VALUE CANIDATE ( NOT SET FOR CONS )           GRA          124     A
GRA       AC       DEFINE 1          ADDRESS CANIDATE                              GRA          125     A
GRA       CON      DEFINE 1          CANDIATE IS A CONSTANT ( FMA, SET )           GRA          126     A
GRA       FP       DEFINE 1          IH IS AN F.P.                                 GRA          127     A
GRA                                                                                GRA          128     A
GRA      *         USEAGE                                                          GRA          129     A
GRA                                                                                GRA          130     A
GRA                 DEFINE 2                                                       GRA          131     A
GRA       EU       DEFINE 1          EXPLICIT USEAGE IN LOOP ( CANNOT BE SUBSUME   GRA          132     A
GRA       RF       DEFINE 1          *RF* ( SHORT ) USEAGE                         GRA          133     A
GRA       ^RD      DEFINE 1          NON RECURSIVE DEFINITION                      GRA          134     A
GRA       IA       DEFINE 1          IA/IS USEAGE                                  GRA          135     A
GRA       TU       DEFINE 1          TEST USEAGE ( IN LOOP TERMINATION TEST )      GRA          136     A
GRA                                                                                GRA          137     A
GRA      *         PROFIT CRITERIA                                                 GRA          138     A
GRA                                                                                GRA          139     A
GRA       PRFT     DEFINE 3          PROFIT                                        GRA          140     A
GRA       APLN     DEFINE 2          APPEARENCE IN ARTICULATION PT/LATCH NODE      GRA          141     A
GRA       NOCC     DEFINE 9          NUMBER OF OCCURANCES                          GRA          142     A
GRA       SUSE     DEFINE 9          SUM( USES ) + SUM( USES(INC)-1 )              GRA          143     A
GRA                                                                                GRA          144     A
GRA      *         ENTRY/EXIT CONDITIONS                                           GRA          145     A
GRA                                                                                GRA          146     A
GRA       PL       DEFINE 1          PRELOAD NECESSARY                             GRA          147     A
GRA       PSP      DEFINE 1          POST STORE POSSIBLE                           GRA          148     A
GRA       DEF      DEFINE 1          DEFINED IN REGION                             GRA          149     A
GRA       USE      DEFINE 1          USED IN REGION                                GRA          150     A
GRA       LX       DEFINE 1          LIVE ON EXIT FROM REGION                      GRA          151     A
GRA       KD       DEFINE 1          KILL DEFS ( STORES )                          GRA          152     A
GRA                                                                                GRA          153     A
GRA       UDI      DEFINE 18         INDEX TO *UDT*                                GRA          154     A
GRA       CAW      SPACE  2                                                        GRA          155     A
GRA       RAF      DEQU   IRA,2      REGISTER ASSIGNMENT FIELDS                    GRA          156     A
GRA       CT       DEQU   CON,3      CANIDATE TYPE  ( CON,AC,VC )                  FCC0594        5     A
GRA       DU       DEQU   USE,2      DEF AND USE BITS                              GRA          157     A
GRA       SORT     DEQU   SUSE,CA.TUP+1-CA.SUSEP  SORT FIELDS                      GRA          158     A
GRA       IH       SPACE  3,14                                                     GRA          159     A
GRA      **        IH. - FORMAT OF SECOND *RCT* WORD                               GRA          160     A
GRA                                                                                GRA          161     A
GRA                 DESCRIBE IH.,60                                                GRA          162     A
GRA       MSK      DEFINE 1          MASK FLAG                                     GRA          163     A
GRA                 DEFINE 5                                                       GRA          164           I
         -CCG16
GRA       SLV      DEFINE 5          SPECIAL LOAD VALUE ( 0/1/2 = LD/LDC,LDV )     CCG16        205     A
GRA       RF       DEFINE 18         RCT INDEX OF *RF* OF REFERENCE                GRA          165     A
GRA       CA       DEFINE 18         BIAS OR CON VALUE OF *S* FOR *JK* OF MASK     GRA          166     A
GRA       IH       DEFINE 18         SYMBOL ORDINAL                                GRA          167     A
GRA       CC       SPACE  2,12                                                     GRA          168     A
GRA      **        CC. - FORMAT OF THIRD WORD OF AN *RCT* ENTRY                    GRA          169     A
GRA                                                                                GRA          170     A
GRA                 DESCRIBE CC.,60                                                GRA          171     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA       I1       DEFINE 1          INCREMENTED ONCE ( FOR A *RD* )                  GRA        172     A
GRA       I2       DEFINE 1          INCREMENTED MORE THAN ONCE                       GRA        173         I
         -FCC0603
GRA       I2       DEFINE 1          INCR MORE THAN ONCE ! INCR ON SIDE BRANCH        FCC0603      9     A
GRA       IM       DEFINE 1          =1 IF INC VALUE IS MINUS VALUE OF *RCT* ENT      GRA        174     A
GRA                DEFINE 3                                                           GRA        175     A
GRA       INC      DEFINE 18         *RCT* INDEX OF INCREMENT VALUE OF FIRST INC      GRA        176     A
GRA                                                                                   GRA        177     A
GRA       H2       DEFINE 18         SYMTAB ORDINAL OF SECOND SYM ( IH-H2 )           GRA        178     A
GRA       REG2     DEFINE 6          REGNO OF SECOND REGISTER ( REG1+-REG2 )          GRA        179     A
GRA       MIT      DEFINE 6          MACHINE INSTRUCTION TYPE FOR ADDRESS CANIDA      GRA        180     A
GRA      *                           0 - LD/ST/STT REG1+CA+IH-H2                      GRA        181     A
GRA      *                           1 - PLD/PST/STT  REG1+CA , CA " 0                GRA        182     A
GRA      *                           2 - SLD/SST/SA  REG1+REG2                        GRA        183     A
GRA      *                           3 - SDL/SDS/SS  REG1-REG2                        GRA        184     A
GRA      *                           4 - SLD/SST/SA  REG1+REG2 , WHERE RJ IS          FCC0646     14     A
GRA      *                                 FROM RF OF OLD INST, NOT *RVT*.            FCC0646     15     A
GRA       V.SRF    EQU    1S11       42 - SIMILAR *IP* ADDRESS DIFFERENCES            FCC0977      8     A
GRA                                                                                   GRA        185     A
GRA       REG1     DEFINE 6          REGNO OF REGISTER ASSIGNED                       GRA        186     A
GRA                                                                                   FCC0811      6     A
GRA       REGF     DEQU   REG1,18    REG1, REG2 AND MIT FIELDS                        FCC0811      7     A
GRA       RAT      SPACE  3,20                                                        GRA        187     A
GRA      **        RA. - REGISTER ASSIGNMENT TABLE FORMAT                             GRA        188     A
GRA      *                                                                            GRA        189     A
GRA      *         *RAT* HOLDS THE RCT ORDINALS OF THE VALUE THAT REGISTERS WILL      GRA        190     A
GRA      *         BE INITIALIZED TO IN THE HOLDING BLOCK ( LOOP PROLOGUE ).          GRA        191     A
GRA      *                                                                            GRA        192     A
GRA      *         THE EVALUATION FORMULA IS - SET(R3) + R1 - R2 .                    GRA        193     A
GRA      *         AND THE *RF* FIELD IN THE *RCT* ENTRY OF R3 IS IGNORED.            GRA        194     A
GRA                                                                                   GRA        195     A
GRA                DESCRIBE RA.,60                                                    GRA        196     A
GRA       INV      DEFINE 1          =1 IF REGISTER IS *INV* IN LOOP                  GRA        197     A
GRA                DEFINE 5                                                           GRA        198     A
GRA       R3       DEFINE 18         RCT ORDINALS                                     GRA        199     A
GRA       R2       DEFINE 18                                                          GRA        200     A
GRA       R1       DEFINE 18                                                          GRA        201     A
GRA                                                                                   GRA        202     A
GRA                LIST   R,-X                                                        GRA        203     A
GRA       GRA      TITLE  DEBUGGING FACILITIES                                        GRA        204     A
GRA      **        PRNT - PRINT OUT SELECTED STRUCTURES                               GRA        205     A
GRA      *                                                                            GRA        206     A
GRA      *         PRNT   LAB,(RCT,RAT,RVT)                                           GRA        207     A
GRA                                                                                   GRA        208     A
GRA       .1       SET    1                                                           GRA        209     A
GRA                ECHO   2,X=(RCT,RAT,RVT,SEE,MTA,DXA,DAA,MFA,SUP,SXC)               GRA        210     A
GRA       Q.X      SET    .1                                                          GRA        211     A
GRA       .1       SET    .1+1                                                        GRA        212     A
GRA                                                                                   GRA        213     A
GRA       PRNT     MACRO  LAB,LIST                                                    GRA        214     A
GRA                LOCAL  APL                                                         GRA        215     A
GRA       O        IF     DEF,DEBUG                                                   GRA        216         I

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
          -FCC1196
GRA        O       IF     DEF,/DEBUG/LAB                                  FCC1196        5     A
GRA       +        RJ     CPR                                             GRA          217     A
GRA       -        VFD    30/APL                                          GRA          218     A
GRA                USE    DEBUG                                           GRA          219     A
GRA        APL     CON    10H LAB                                         GRA          220     A
GRA                IF     DEF,Q.LAB,1                                     GRA          221     A
GRA                VFD    6/Q.LAB                                         GRA          222     A
GRA                IF     DEF,/DEBUG/LAB,3                                GRA          223         I
          -FCC1196
GRA                IRP    LIST                                            GRA          224     A
GRA                VFD    6/Q.LIST                                        GRA          225     A
GRA                IRP                                                    GRA          226     A
GRA                VFD    *P/0                                            GRA          227     A
GRA                USE    *                                               GRA          228     A
GRA        O       ENDIF                                                  GRA          229     A
GRA                ENDM                                                   GRA          230     A
GRA                                                                       GRA          231     A
GRA        .T      IFNE   TEST,0                          *TEST MODE*     GRA          232     A
GRA                                                                       GRA          233     A
GRA        CPR     ROUTINE                                                GRA          234     A
GRA                RJ     =XSAVE=                                         GRA          235         I
          -CCG7
GRA                RJ     =XSVR=                                          CCG7         113     A
GRA                SA1    CPR                                             GRA          236     A
GRA                SA3    O.RCT                                           GRA          237     A
GRA                LX1    30                                              GRA          238     A
GRA                SA2    X1-1                                            GRA          239     A
GRA                SX6    X2                                              GRA          240     A
GRA                BX7    X3                                              GRA          241     A
GRA                SA6    CPRA        LOC(LAB)                            GRA          242     A
GRA                SA7    CPRA+1                                          GRA          243     A
GRA                SX6    X6+1                                            GRA          244     A
GRA                SA6    CPRA+3      LOC(INDEX)                          GRA          245     A
GRA                SA1    CPRA                                            GRA          246     A
GRA                CALL   PRNTGRA                                         GRA          247     A
GRA                RJ     =XRESET=                                        GRA          248         I
          -CCG7
GRA                RJ     =XRSR=                                          CCG7         114     A
GRA                EQ     CPR                                             GRA          249     A
GRA                                                                       GRA          250     A
GRA        CPRA    BSS    2                                               GRA          251     A
GRA                VFD    42/,18/L.RCT                                    GRA          252     A
GRA                BSSZ   2                                               GRA          253     A
GRA                                                                       GRA          254     A
GRA                USE    /RCTFMT/                                        GRA          255     A
GRA                ECHO   1,X=(RA,VC,AC,CON,FP,EU,RF,^RD,IA,TU,PL,PSP,DEF,USE,LX,  GRA  256     A
GRA       ,KD,IRA)                                                        GRA          257     A
GRA                VFD    42/0A_X,18/60-CA.X_P                            GRA          258     A
GRA                USE    0                                               GRA          259     A
GRA        .T      ENDIF                                  *TEST MODE*     GRA          260     A
GRA        GRA     TITLE  GRA - CONTROL                                   GRA          261     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA      **        GRA - GLOBAL REGISTER ASSIGNMENT ( CONTROL )                    GRA        262    A
GRA                                                                                FCC0603     10    A
GRA       MARA     DATA   4          MAX A-REGISTER ASSIGNMENTS                    FCC0603     11    A
GRA       GRASRF   ENTRY. 0           =1 IF ADDR DIFFERENCING OF SIMILAR *IP"S*     FCC0977      9          I
         -CCGNC
GRA       GRASRF   ENTRY. 0,#         =1 IF ADDRESS DIFFERENCING OF SIMILAR *IP"S*  CCGNC      646    A
GRA                                                                                GRA        263    A
GRA       GRA      ENTRY.                                                           GRA        264          I
         -CCGNC
GRA       GRA      ENTRY. **,#                                                      CCGNC      647    A
GRA                SX7    0                                                         CCG0022      5    A
GRA                SA7    =XNBIP#     NBIP = 0    */ FOR BDT                        CCG0022      6    A
GRA                RJ     IRP         INITIALIZE REGION PROCESSING                  GRA        265          I
         -FCC0603
GRA                SX7    4                                                         FCC0603     12    A
GRA                SA7    MARA        MARA = 4                                      FCC0603     13    A
GRA                                                                                FCC0603     14    A
GRA       GRA0     RJ     IRP         INITIALIZE REGION PROCESSING                  FCC0603     15    A
GRA                IF     DEF,/DEBUG/UDT,1                                          GRA        266    A
GRA                DCALL  PRNTUDI,([O.UDT],L.UDT)                                   GRA        267    A
GRA                                                                                GRA        268    A
GRA                SA1    MRA                                                       GRA        269    A
GRA                ZR     X1,GRA3     IF MRA = 0    */ USER OR I/O EXT REFS         GRA        270    A
GRA                                                                                GRA        271    A
GRA                CLCM   DUM         DETERMINE USEAGE MODE                         GRA        272    A
GRA                CLCM   ERC         ENTER REGISTER CANIDATES                      GRA        273    A
GRA                SA5    L.RCT                                                     GRA        274    A
GRA                SX6    X5+         LCT = L.RCT                                   GRA        275    A
GRA                SA6    LCT                                                       GRA        276    A
GRA                                                                                GRA        277    A
GRA                RJ     SEE         SET ENTRY/EXIT CONDITIONS FOR VALUE CANIDAT   GRA        278    A
GRA                PRNT   SEE                                                       GRA        279    A
GRA                                                                                GRA        280    A
GRA                RJ     MTA         MAKE TENATIVE B-ASSIGNMENTS, COUNTING METHO   GRA        281    A
GRA                PRNT   MTA,(RAT,RCT)                                             GRA        282    A
GRA                                                                                GRA        283    A
GRA                SA1    XRF                                                       GRA        284    A
GRA                NZ     X1,GRA2     IF XRF " 0    */ EXT REFS IN THE LOOP         GRA        285    A
GRA                SA2    N.HB                                                      GRA        286    A
GRA                NZ     X2,GRA1     IF N.HB " 0   */ NOT INNERMOST LOOP           GRA        287    A
GRA                                                                                GRA        288    A
GRA                SA3    MAXW                                                      GRA        289    A
GRA                SA4    NTA                                                       GRA        290    A
GRA                SB3    X3+                                                       GRA        291    A
GRA                SB4    X4+8                                                      GRA        292    A
GRA                GT     B3,B4,GRA2  IF MAXW > NTA+8 */ LOOP IS TOO WIDE           GRA        293    A
GRA                                                                                GRA        294    A
GRA                RJ     DXA         DETERMINE X-ASSIGNMENTS                       GRA        295    A
GRA                PRNT   DXA                                                       GRA        296    A
GRA                                                                                GRA        297    A
GRA                RJ     DAA         DETERMINE A-ASSIGNMENTS                       GRA        298    A
GRA                PRNT   DAA,(RAT,RCT)                                             GRA        299          I

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

          -FCC1039
GRA                 PRNT   DAA                                                  FCC1039        6     A
GRA                 EQ     GRA2                                                 GRA          300     A
GRA                                                                             GRA          301     A
GRA       GRA1      RJ     DXA         MOVE PREVIOUS X-ASSIGNMENTS OUT          GRA          302     A
GRA                                                                             GRA          303     A
GRA       GRA2      RJ     MFA         MAKE FINAL B-ASSIGNMENTS                 GRA          304     A
GRA                 PRNT   MFA,(RAT,RCT)                                        GRA          305     A
GRA                                                                             GRA          306     A
GRA       GRA3      RJ     CLB         CODE LOOP BODY                           GRA          307     A
GRA                                                                             GRA          308     A
GRA                 RJ     SUP         SETUP PRELOADS IN HOLDING BLOCK          GRA          309     A
GRA                 PRNT   SUP,RAT                                              GRA          310     A
GRA                                                                             GRA          311     A
GRA                 RJ     SXC         SET EXIT CONDITIONS ( POST STORES, ETC. ) GRA         312     A
GRA                 TRACE  SXC,TET                                              GRA          313     A
GRA                 PRNTABV  SXC,(LUV,LEA)                                      GRA          314     A
GRA                 SX6    0                                                    GRA          315     A
GRA                 SA6    L.RCT       L.RCT = 0                                GRA          316     A
GRA                 SA6    L.RXI       L.RXI = 0                                GRA          317     A
GRA                 SA6    AXCT        AXCT = 0                        000840   CCG34          9         I
          -CCG0057
GRA                 SA6    GRASRF      GRASRF = 0                               FCC0977       10     A
GRA                 EQ     GRA                                                  GRA          318     A
GRA       IRP       TITLE  IRP - INITIALIZE REGION PROCESSING                   GRA          319         I
          -FCC0603
GRA       IRP       SPACE  3                                                    FCC0603       16     A
GRA       **        IRP - INITIALIZE REGION PROCESSING                          GRA          320     A
GRA                                                                             GRA          321     A
GRA       IRP       ROUTINE                                                     GRA          322     A
GRA                                                                             GRA          323     A
GRA                 SETZERO F.CLC,L.CLC      ZERO SCRATCH AREA                  GRA          324     A
GRA                                                                             GRA          325     A
GRA       *         INITIALIZE *RCT* , FIRST ENTRY IS AN EMPTY ONE, USED AS A  GRA          326     A
GRA       *         SEARCH TERMINATOR, SECOND ENTRY IS THE CONSTANT *1* .      GRA          327     A
GRA                                                                             GRA          328     A
GRA                 SA5    IRPA                                                 GRA          329     A
GRA                 ALLOC  RCT,6       ALLOC( RCT , 6 )                         GRA          330     A
GRA                 MX6    0                                                    GRA          331     A
GRA                 SA6    AXCT        AXCT = 0                                 CCG0057        6     A
GRA                 SA6    X2                                                   GRA          332     A
GRA                 SA6    A6+B1                                                GRA          333     A
GRA                 SA6    A6+B1                                                GRA          334     A
GRA                 BX7    X5                                                   GRA          335     A
GRA                 SA7    A6+B1       RCT(4) = CAW(0,CON,RF,0)  */ CON 1       GRA          336     A
GRA                 SX7    B1                                                   GRA          337     A
GRA                 LX7    IH.CAP                                               GRA          338     A
GRA                 SA7    A7+B1       RCT(5) = IHW(0,0,1,0)                    GRA          339     A
GRA                 SA6    A7+B1       RCT(6) = 0                               GRA          340     A
GRA                 SX7    100                                                  GRA          341     A
GRA                 SA7    TUD         TUD = 100                                GRA          342     A
GRA                                                                             GRA          343     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SA1    N.HB                                                    GRA          344     A
GRA                ZR     X1,IRP1     IF N.HB = 0   */ INNERMOST LOOP             GRA          345     A
GRA                SA4    LUV                                                     GRA          346     A
GRA                SX5    B1                                                      GRA          347     A
GRA                LX5    UD.ISTP                                                 GRA          348     A
GRA                CALL   EBV         EBV( LUV , IST )  */ SET *IST* FOR DUM/ERC  GRA          349         I
          -CCGNC
GRA                CALL   EBV#        EBV( LUV , IST )  */ SET *IST* FOR DUM/ERC  CCGNC        648     A
GRA                                                                               GRA          350     A
GRA       *        ADJUST *TET*, CLEAR *REG* FIELD, UPDATE INFO OF *EQV* ENTRIES  GRA          351         I
          -FCC0514
GRA                                                                               GRA          352         I
          -FCC0514
GRA        IRP1    SA2    ITL                                                     GRA          353         I
          -FCC0514
GRA                SA3    =XN.GT                                                  GRA          354         I
          -FCC0514
GRA                SA1    O.TET                                                   GRA          355         I
          -FCC0514
GRA                SB2    X2          I = ITL                                     GRA          356         I
          -FCC0514
GRA                SB3    X3          L = N.GT                                    GRA          357         I
          -FCC0514
GRA                EQ     B2,B3,IRP3  IF ITL = N.GT */ NO TET"S CREATED           GRA          358         I
          -FCC0514
GRA                                                                               GRA          359         I
          -FCC0514
GRA                MX0    -T.REGL                                                 GRA          360         I
          -FCC0514
GRA                LX0    T.REGP                                                  GRA          361         I
          -FCC0514
GRA        IRP2    SA2    X1+B2                                                   GRA          362         I
          -FCC0514
GRA                BX6    X0*X2       REG[TET(I)] = 0                             GRA          363         I
          -FCC0514
GRA                SB2    B2+B1       I = I + 1                                   GRA          364         I
          -FCC0514
GRA                SA6    A2                                                      GRA          365         I
          -FCC0514
GRA                LT     B2,B3,IRP2  IF I < L                                    GRA          366         I
          -FCC0514
GRA                                                                               GRA          367         I
          -FCC0514
GRA        IRP3    SA4    L.TET                                                   GRA          368         I
          -FCC0514
GRA                SB5    X1                                                      GRA          369         I
          -FCC0514
GRA                SB4    X4          I = L.TET                                   GRA          370         I
          -FCC0514
GRA                EQ     B4,B3,IRP   IF L.TET = N.GT  */ NO TST SQUEEZING        GRA          371         I
          -FCC0514
GRA                                                                               GRA          372         I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
          -FCC0514
GRA                 MX0    -T.IPFL                                          GRA          373          I
          -FCC0514
GRA                 SX6    B3          L.TET = N.GT                         GRA          374          I
          -FCC0514
GRA                 LX0    T.IPFP                                           GRA          375          I
          -FCC0514
GRA                 MX1    1+T.CAL                                          GRA          376          I
          -FCC0514
GRA                 SA6    A4                                               GRA          377          I
          -FCC0514
GRA                 LX1    T.CAL                                            GRA          378          I
          -FCC0514
GRA                                                                         GRA          379          I
          -FCC0514
GRA        IRP4     SB4    B4-B1       I = I - 1                            GRA          380          I
          -FCC0514
GRA                 SA2    B5+B4       TI = TET(I)                          GRA          381          I
          -FCC0514
GRA                 BX7    X1*X2       C = CA[TI]                           GRA          382          I
          -FCC0514
GRA                 SA3    B5+X2                                            GRA          383          I
          -FCC0514
GRA                 BX4    -X0*X3      IPI = IPF[TET(C)]                    GRA          384          I
          -FCC0514
GRA                 IX6    X4+X7       TET(C) = TETW(INV[TI],IPI,0,C)       GRA          385          I
          -FCC0514
GRA                 SA6    A3+                                              GRA          386          I
          -FCC0514
GRA                 GT     B4,B3,IRP4  IF I > L                             GRA          387          I
          -FCC0514
GRA                                                                         GRA          388          I
          -FCC0514
GRA                 EQ     IRP                                              GRA          389          I
          -FCC0514
GRA        IRP1     EQ     IRP                                              FCC0514       97          I
          -FCC0909
GRA       *         RESERVE *B3* FOR ASSIGNED GOTO COMPARE TEST, IF A       FCC0909       30     A
GRA       *         LABEL SWITCH WAS PERFORMED ( LGL " 0 ) .                FCC0909       31     A
GRA                                                                         FCC0909       32     A
GRA        IRP1     SA1    N.LJ                                             FCC0909       33     A
GRA                 LX1    1                                                FCC0909       34     A
GRA                 PL     X1,IRP      IF ^B58[N.LJ]                        FCC0909       35     A
GRA                                                            A            FCC0909       36     A
GRA                 SA2    MRA                                              FCC0909       37     A
GRA                 SX7    1S3                                              FCC0909       38     A
GRA                 BX6    -X7*X2      MRA = MRA & ^1S3  */ MAKE *B3* UNAVAIL FCC0909     39     A
GRA                 SA6    A2                                               FCC0909       40     A
GRA                 SA7    TRA         TRA = 1S3     */ INDICATE TEMP USE OF *B3* FCC0909 41     A
GRA                 EQ     IRP                                              FCC0909       42     A
GRA                                                                         GRA          390     A
GRA        IRPA     BFMW   CA,(CON,RF)                                      GRA          391     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA       GRA      TTL    GRA - GLOBAL REGISTER ASSIGNMENT / CANIDATE ENTRY                GRA          392     A
GRA       SEE      TITLE  SEE - SET ENTRY/EXIT CONDITIONS                                  GRA          393     A
GRA      **        SEE - SET ENTRY/EXIT CONDITIONS FOR VALUE CANIDATES IN *RCT*            GRA          394     A
GRA                                                                                        GRA          395     A
GRA       SEE      ROUTINE                                                                 GRA          396     A
GRA                SA5    O.RCT                                                            GRA          397     A
GRA                SA4    L.RCT                                                            GRA          398     A
GRA                SA3    BVL                                                              GRA          399     A
GRA                SA2    LEA                                                              GRA          400     A
GRA                SA1    O.UDT                                                            GRA          401     A
GRA                SB2    X4                                                               GRA          402     A
GRA                SB7    X2                                                               GRA          403     A
GRA                MX0    -CA.DUL                                                          GRA          404     A
GRA                SB6    X3+B1       VL = BVL + 1  */ BIT VECTOR LENGTH                   GRA          405     A
GRA                SB5    59-CA.VCP                                                        GRA          406     A
GRA                SA5    X5+B2       CI = O.RCT + L.RCT                                   GRA          407     A
GRA                SB3    3                                                                GRA          408     A
GRA                SA0    X1+B1       UB = O.UDT + 1                                       GRA          409     A
GRA                SB4    B0          NV = 0                                               GRA          410     A
GRA                EQ     SEE2                                                             GRA          411     A
GRA                                                                                        GRA          412     A
GRA      *         COMPILER TEMPORARY, SET PROFIT = DU[CI]+1                               GRA          413     A
GRA                                                                                        GRA          414     A
GRA       SEE1     SX7    X7+B1                                                            GRA          415     A
GRA                LX7    CA.PRFTP                                                         GRA          416     A
GRA                SX4    B1                                                               GRA          417     A
GRA                BX6    X7+X5       PRFT[CI] = DU[CI] + 1                                GRA          418     A
GRA                LX4    CA.PSPP                                                          GRA          419     A
GRA                BX6    X4+X6       PSP[CI] = 1                                          GRA          420     A
GRA                SA6    A5+                                                              GRA          421     A
GRA                                                                                        GRA          422     A
GRA      *         SEARCH BACKWARDS FOR VALUE CANIDATES                                    GRA          423     A
GRA                                                                                        GRA          424     A
GRA       SEE2     SA5    A5-B3       CI = CI - 3                                          GRA          425     A
GRA                ZR     X5,SEE6     IF [CI] = 0   */ END OF TABLE                        GRA          426     A
GRA                LX6    B5,X5                                                            GRA          427     A
GRA                PL     X6,SEE2     IF ^VC[CI]                                           GRA          428     A
GRA                                                                                        GRA          429     A
GRA                SB2    X5          I = UDI[CI]                                          GRA          430     A
GRA                LX6    1+CA.VCP-CA.DUP                                                  GRA          431     A
GRA                BX7    -X0*X6      DUB = DU[CI]                                         GRA          432     A
GRA                SB4    B4+B1       NV = NV + 1                                          GRA          433     A
GRA                ZR     B2,SEE1     IF I = 0      */ COMPILER TEMP                       GRA          434     A
GRA                GE     B2,B3,SEE2A IF I > 2      */ PROGRAMMER VAR                      FCC0645        9     A
GRA                                                                                        FCC0645       10     A
GRA      *         LOAD ONLY VARIABLE, SET PRFT = 1 AND PL = 1                             FCC0645       11     A
GRA                                                                                        FCC0645       12     A
GRA                SX4    B1                                                               FCC0645       13     A
GRA                LX4    CA.PLP      PL[CI] = 1                                           FCC0645       14     A
GRA                BX5    X4+X5                                                            FCC0645       15     A
GRA                LX4    CA.PRFTP-CA.PLP                                                  FCC0645       16     A
GRA                BX6    X4+X5       PRFT[CI] = 1                                         FCC0645       17     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SA6    A5                                                    FCC0645      18     A
GRA                EQ     SEE2                                                  FCC0645      19     A
GRA                                                                             GRA         435     A
GRA       *        PROCESS PROGRAMMER VARIABLE, EXIT INFO COMPUTED IN *FXI*     GRA         436     A
GRA                                                                             GRA         437     A
GRA                SA4    A0+B2       U2 = UDT(I+1)                             GRA         438          I
          -FCC0645
GRA        SEE2A   SA4    A0+B2       U2 = UDT(I+1)                             FCC0645      20     A
GRA                SA2    B7+X4       VA = LEA + WI[UI];  LE = [VA]             GRA         439     A
GRA                SX1    B1                                                    GRA         440     A
GRA                UX3    B2,X4       B = BITN[U2]                              GRA         441     A
GRA                AX4    B2,X2                                                 GRA         442     A
GRA                BX2    X1*X4                                                 GRA         443     A
GRA                LX2    CA.PLP      PL[CI] = SHIFT(LE,-B)&1  */ PRELOAD       GRA         444     A
GRA                BX5    X2+X5                                                 GRA         445     A
GRA                SA3    A2-B6       LXA = VA - VL;  LX = [LXA]                GRA         446     A
GRA                AX4    B2,X3                                                 GRA         447     A
GRA                BX2    X1*X4                                                 GRA         448     A
GRA                LX2    CA.LXP      LX[CI] = SHIFT(LX,-B)&1  */ LIVE EXIT     GRA         449     A
GRA                BX5    X2+X5                                                 GRA         450     A
GRA                SA3    A2+B6       VA = VA + VL;  MD = [VA]  */ MOVABLE DEFS  GRA         451     A
GRA                AX4    B2,X3                                                 GRA         452     A
GRA                BX1    X1*X4                                                 GRA         453     A
GRA                LX1    CA.PSPP     PSP[CI] = SHIFT(MD,-B)&1  */ POST ST POSSIB GRA        454     A
GRA                BX5    X1+X5                                                 GRA         455     A
GRA                                                                             GRA         456     A
GRA                LX6    CA.DUP+59-CA.DEFP                                     GRA         457     A
GRA                PL     X6,SEE3     IF ^DEF[CI]                               GRA         458     A
GRA                NZ     X1,SEE3     IF PSP[CI]    */ POST STORE POSSIBLE      GRA         459     A
GRA                BX4    X6                                                    GRA         460     A
GRA                LX4    CA.DEFP-CA.TUP                                        GRA         461     A
GRA                MI     X4,SEE3     IT TU[CI]     */ KEEP CV FIRST IN MTA SORT GRA        462     A
GRA                SX7    X7-2        DUB = DUB - 2                             GRA         463     A
GRA                NZ     X7,SEE3     IF DUB " 0    */ USED ALSO IN REGION      GRA         464     A
GRA                                                                             GRA         465     A
GRA       *        DEF ONLY AND NOT MOVABLE, CLEAR VC BIT                       GRA         466     A
GRA                                                                             GRA         467     A
GRA                SX1    B1                                                    GRA         468     A
GRA                LX1    CA.VCP                                                GRA         469     A
GRA                BX6    -X1*X5      VC[CI] = 0                                GRA         470     A
GRA                SA6    A5                                                    GRA         471     A
GRA                EQ     SEE2                                                  GRA         472     A
GRA                                                                             GRA         473     A
GRA       *        SET PROFIT = 1 IF USE ! DEF , ELSE 3                         GRA         474     A
GRA                                                                             GRA         475     A
GRA        SEE3    SB2    X7+B1                                                 GRA         476     A
GRA                NE     B2,B3,SEE4  IF DUB " 2    */ NOT DEF ONLY             GRA         477     A
GRA                SX7    B1          DUB = 1                                   GRA         478     A
GRA                                                                             GRA         479     A
GRA        SEE4    LX7    CA.PRFTP                                              GRA         480     A
GRA                BX7    X5+X7       PRFT[CI] = DUB                            GRA         481     A
GRA                LX6    CA.DEFP-CA.^RDP                                       GRA         482          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

          -FCC0839
GRA                 SA7    A5                                                GRA          483          I
          -FCC0839
GRA                 SA7    A5+                                               FCC0839        4     A
GRA                 PL     X6,SEE2     IF ^DEF[CI]    */ USED ONLY           FCC0839        5     A
GRA                 LX6    CA.DEFP-CA.^RDP                                   FCC0839        6     A
GRA                 NZ     X2,SEE2     IF LX[CI]                             GRA          484     A
GRA                                                                         GRA          485     A
GRA       *         DEAD ON EXIT FROM THE LOOP, KILL DEFINITIONS OF CANIDATE GRA          486     A
GRA       *         IF IT HAS NO REAL USES IN THE LOOP                      GRA          487     A
GRA                                                                         GRA          488     A
GRA                 PL     X6,SEE4A    IF RD[CI]                             GRA          489     A
GRA                                                                         GRA          490     A
GRA                 LX6    CA.^RDP-CA.USEP                                   GRA          491     A
GRA                 MI     X6,SEE2     IF USE[CI]                            GRA          492     A
GRA                                                                         GRA          493     A
GRA                 MX6    0           DIFF = 0                              GRA          494     A
GRA                 EQ     SEE5                                              GRA          495     A
GRA                                                                         GRA          496     A
GRA        SEE4     SA4    A4-B1       UI = UDT(I)                           GRA          497     A
GRA                 LX5    -CA.SUSEP                                         GRA          498     A
GRA                 LX4    -UD.STCP                                          GRA          499     A
GRA                 MX1    -UD.STCL                                          GRA          500     A
GRA                 BX4    -X1*X4                                            GRA          501     A
GRA                 BX1    -X1*X5                                            GRA          502     A
GRA                 IX6    X1-X4       DIFF = SUSE[CI] - STC[UI]             GRA          503     A
GRA                 LX5    CA.SUSEP+59-CA.TUP                                GRA          504     A
GRA                 PL     X5,SEE5     IF ^TU[CI]    */ NOT THE LOOP *CV*    GRA          505     A
GRA                                                                         GRA          506     A
GRA                 SA6    TUD         TUD = DIFF    */ SAVE *DIFF* FOR LATER USE GRA   507     A
GRA                 EQ     SEE2                                              GRA          508     A
GRA                                                                         GRA          509     A
GRA        SEE5     NZ     X6,SEE2     IF DIFF " 0   */ VAR HAS REAL USES    GRA          510     A
GRA                                                                         GRA          511     A
GRA                 SA4    SEEA                                              GRA          512     A
GRA                 BX7    X4+X7       (RA,KD)[CI] = 1                       GRA          513     A
GRA                 SA7    A7                                                GRA          514     A
GRA                 EQ     SEE2                                              GRA          515     A
GRA        SEE      SPACE  2,14                                              GRA          516     A
GRA       *         CHECK FOR FINAL ASSIGNMENTS OF CANIDATES IN AN OUTER LOOP GRA          517     A
GRA       *         AND IF SO, ADJUST *INV* BIT OF *RAT* ENTRIES.           GRA          518     A
GRA                                                                         GRA          519     A
GRA        SEE6     SA1    NXC                                               GRA          520     A
GRA                 SA2    FBA                                               GRA          521     A
GRA                 SX6    B4+B1       NVC = NV + 1  */ N. VALUE CANIDATES   GRA          522     A
GRA                 SB6    A5                                                GRA          523     A
GRA                 SA6    NVC                                               GRA          524     A
GRA                 IX7    X1+X2                                             GRA          525     A
GRA                 ZR     X7,SEE11    IF FBA + NXC = 0  */ NO FINAL ASSIGNMENTS GRA      526     A
GRA                 MX0    1                                                 GRA          527     A
GRA                 SX2    27B         I = 27B                               GRA          528     A
GRA                 SB7    RAT                                               GRA          529     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                NZ     X1,SEE7     IF NXC " 0                                    GRA          530     A
GRA                SX2    7           I = 7                                         GRA          531     A
GRA                                                                                 GRA          532     A
GRA        SEE7    SA5    B7+X2                                                     GRA          533     A
GRA                ZR     X5,SEE8     IF RAT(I) = 0                                 GRA          534     A
GRA                SA4    B6+X5       J = R1[RAT(I)]                                GRA          535         I
           -FCC1014
GRA                SA4    B6+X5       J = R1[RAT(I)];  CAW = RCT(J)                 FCC1014        5     A
GRA                LX4    59-CA.DEFP                                                GRA          536     A
GRA                BX3    -X4*X0                                                    GRA          537     A
GRA                LX3    1+RA.INVP                                                 GRA          538     A
GRA                BX6    X3+X5       INV[RAT(I)] = ^DEF[RCT(J)]                    GRA          539     A
GRA                SA6    A5                                                        GRA          540     A
GRA                LX4    CA.DEFP-CA.RAP                                            GRA          541     A
GRA        .FTN    IFEQ   HC.ID,2                                  ?FTN             CCG4          60     A
GRA                MI     X4,SEE8     IF RA[RCT(J)]                                 GRA          542         I
           -FCC1014
GRA                PL     X4,SEE7A    IF ^RA[CAW]                                   FCC1014        6     A
GRA                                                                                 FCC1014        7     A
GRA        *       CHECK FOR CANDIDATE OF FORM  LD  A0+K AND CHANGE TO F.P. ADDR    FCC1014        8     A
GRA                                                                                 FCC1014        9     A
GRA                AX5    RA.R1L                                                    FCC1014       10     A
GRA                NZ     X5,SEE8     IF (R2,R3)[CAW] " 0                           FCC1014       11     A
GRA                LX4    1+CA.RAP                                                  FCC1014       12     A
GRA                SX6    X4-2                                                      FCC1014       13     A
GRA                NZ     X6,SEE8     IF UDI[CAW] " 1                               FCC1014       14     A
GRA                SA3    A4+B1       C2W = RCT(J+1)                                FCC1014       15     A
GRA                LX3    -IH.RFP                                                   FCC1014       16     A
GRA                SB2    X3                                                        FCC1014       17     A
GRA                LX3    IH.RFP-IH.CAP                                             FCC1014       18     A
GRA                NE     B2,B1,SEE8  IF RF[C2W] " 1                                FCC1014       19     A
GRA                                                                     A           FCC1014       20     A
GRA                SX6    X3+2                                                      FCC1014       21         I
           -CCG4
GRA                SA6    A3          RCT(J+1) = IHW(0,0,CA[C2W]+2)                 FCC1014       22         I
           -CCG4
GRA                SA1    =XO.FPT                                                   CCG4          61         I
           -CCG18
GRA                SB2    X3                                                        CCG4          62         I
           -CCG18
GRA                SA3    X1+B2                                                     CCG4          63         I
           -CCG18
GRA                SX6    X3                                                        CCG4          64         I
           -CCG18
GRA                SA6    A3          RCT(J+1) = IHW(0,0,FPT(CA[C2W])               CCG4          65         I
           -CCG18
GRA                SA1    =XO$FPI                                                   CCG18        101     A
GRA                SB2    X3                                                        CCG18        102     A
GRA                SA1    X1+B2                                                     CCG18        103     A
GRA                LX1    -FP.PNTP    IH = PNT[FPI(CA)]                             CCG18        104     A
GRA                SX6    X1                                                        CCG18        105     A
GRA                SA6    A3          RCT(J+1) = IHW(0,0,IH)                        CCG18        106     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SX3    B1                                                    FCC1014      23     A
GRA                LX3    CA.ACP                                                FCC1014      24     A
GRA                BX6    X4+X3        AC[CAW] = 1                              FCC1014      25     A
GRA                LX3    CA.VCP-CA.ACP                                         FCC1014      26     A
GRA                BX6    -X3*X6       VC[CAW] = 0                              FCC1014      27     A
GRA                SA6    A4                                                    FCC1014      28     A
GRA                SA4    A4+2                                                  FCC1014      29     A
GRA                SX5    2S6                                                   FCC1014      30     A
GRA                BX6    X4+X5        MIT[RCT(J+2)] = 2                        FCC1014      31     A
GRA                SA6    A4                                                    FCC1014      32     A
GRA                EQ     SEE8                                                  FCC1014      33     A
GRA       .FTN     ENDIF                                          'FTN          CCG4         66     A
GRA                                                                             GRA         543     A
GRA      *         INV ADDRESS CANIDATE AND *RF* NOT MARKED                      GRA         544     A
GRA                                                                             GRA         545     A
GRA                LX4    1+CA.RAP-CA.SUSEP                                     GRA         546           I
          -FCC1014
GRA       SEE7A    LX4    1+CA.RAP-CA.SUSEP                                     FCC1014      34     A
GRA                MX7    -CA.SUSEL                                             GRA         547     A
GRA                BX3    -X7*X4                                                GRA         548     A
GRA                NZ     X3,SEE8      IF SUSE[RCT(J)] " 0  */ OTHER USES       GRA         549     A
GRA                                                                             GRA         550     A
GRA                LX4    CA.SUSEP+59-CA.RAP                                    GRA         551     A
GRA                BX6    X0+X4        RA[RCT(J)] = 1                           GRA         552     A
GRA                LX6    1+CA.RAP                                              GRA         553     A
GRA                SX7    B1                                                    GRA         554     A
GRA                LX7    CA.RFP                                                GRA         555     A
GRA                BX6    -X7*X6       RF[RCT(J)] = 0  */ FOR *MTA*             GRA         556     A
GRA                SA6    A4                                                    GRA         557     A
GRA                                                                             GRA         558     A
GRA       SEE8     SX2    X2-1         I = I - 1                                GRA         559     A
GRA                NZ     X2,SEE7      IF I " 0                                 GRA         560     A
GRA       SEE      SPACE  1,10                                                  GRA         561     A
GRA      *         SCAN *IA* LIST TO SET USEAGE MODE FOR ENTRIES ON IT           GRA         562     A
GRA                                                                             GRA         563     A
GRA       SEE11    SA1    IAI                                                   GRA         564     A
GRA                ZR     X1,SEE       IF IAI = 0    */ NO *IA* INCRS           GRA         565     A
GRA                                                                             GRA         566     A
GRA                SB2    SCR          K = 0                                    GRA         567     A
GRA                SB3    X1+B2                                                 GRA         568     A
GRA                SX0    B1                                                    GRA         569     A
GRA                LX0    CA.RFP                                                GRA         570     A
GRA                SX1    B1                                                    GRA         571     A
GRA                LX1    CA.EUP                                                GRA         572     A
GRA                SB5    59-CA.KDP                                             GRA         573     A
GRA                SX4    B1                                                    FCC0535       5     A
GRA                LX4    CA.LXP                                                FCC0535       6     A
GRA                                                                             GRA         574     A
GRA       SEE12    SA2    B2           (J,I) = SCR(K)                           GRA         575     A
GRA                SB2    B2+B1        K = K + 1                                GRA         576     A
GRA                SA5    B6+X2        CAW = RCT(I)  */ *RD* VAR                GRA         577     A
GRA                LX6    B5,X5                                                 GRA         578     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                MI     X6,SEE14    IF KD[CAW]    */ INC IS USELESS                GRA          579     A
GRA                                                                                  GRA          580     A
GRA                BX7    X0*X5       BIT = RF[CAW]                                   GRA          581     A
GRA                AX2    18                                                          GRA          582     A
GRA                NZ     X7,SEE13    IF BIT " 0    */ SHORT USEAGE IN THE LOOP       GRA          583     A
GRA                                                                                  GRA          584     A
GRA                BX6    X1+X5       EU[CAW] = 1   */ MARK AS LONG USEAGE            GRA          585     A
GRA                SA6    A5                                                          GRA          586     A
GRA                BX7    X1          BIT = EU                                        GRA          587     A
GRA                                                                                  GRA          588     A
GRA       SEE13    SA3    B6+X2                                                       GRA          589     A
GRA                BX6    X7+X3       BIT[RCT(J)] = 1                                 GRA          590     A
GRA                SA6    A3+                                                         GRA          591           I
          -FCC0535
GRA                SA6    A3                                                          FCC0535        7     A
GRA                LX3    59-CA.CONP                                                  FCC0535        8     A
GRA                MI     X3,SEE14    IF CON[RCT(J)]                                  FCC0535        9     A
GRA                BX6    X4+X6       LX[RCT(J)] = 1  */ INHIBIT BIASING IN *MFA*     FCC0535       10     A
GRA                SA6    A3                                                          FCC0535       11     A
GRA                                                                                  GRA          592     A
GRA       SEE14    LT     B2,B3,SEE12 IF K < IAI                                      GRA          593     A
GRA                EQ     SEE                                                         GRA          594     A
GRA                                                                                  GRA          595     A
GRA       SEEA     BFMW   CA,(RA,VC,KD)                                               GRA          596     A
GRA       ERC      TITLE  ERC - ENTER REGISTER CANIDATES INTO *RCT*                   GRA          597     A
GRA      **        ERC - ENTER REGISTER CANIDATES INTO *RCT*                          GRA          598     A
GRA      *         FOWARD SCAN OF THE BLOCKS IN THE LOOP TO ENTER CANIDATES           GRA          599     A
GRA      *         BY CALLING *SCT*. LINK WORDS OF CANIDATES POINT TO THEIR *RCT*     GRA          600     A
GRA      *         ENTRIES, ALL OTHERS ARE CLEARED.                                   GRA          601     A
GRA      *         SET BITS IN *LUV* ( LOOP USEAGE VECTOR ) FOR NON CANIDATE          GRA          602     A
GRA      *         MEMORY REFERENCES.                                                 GRA          603     A
GRA      *         MARK FINAL B AND X ASSIGNMENTS FROM INNER LOOPS ( *RS* )           GRA          604     A
GRA                                                                                  GRA          605     A
GRA                QUAL   ERC                                                         GRA          606     A
GRA                                                                                  GRA          607     A
GRA                PROCESS EOQ                                                        GRA          608     A
GRA                SX6    B6-B5       L.RCT = CL - CO                                 GRA          609     A
GRA                SA6    L.RCT                                                       GRA          610     A
GRA                TRACE  ERC,RLIST,SEQ                                               GRA          611     A
GRA                                                                                  GRA          612     A
GRA       ERC      ROUTINE                                                            GRA          613     A
GRA                LX3    B1,X4                                                       GRA          614     A
GRA                IX1    X3+X4                                                       GRA          615     A
GRA                ALLOC  RCT,X1      ALLOC( RCT , 3*NC )                             GRA          616     A
GRA                SB5    X2          CO = O.RCT                                      GRA          617     A
GRA                SA5    O.SEQ                                                       GRA          618     A
GRA                SB6    X2+B6       CL = CO + OLD(L.RCT)                            GRA          619     A
GRA                SA4    LUV                                                         GRA          620     A
GRA                SA3    O.UDT                                                       GRA          621     A
GRA                SB3    3                                                           GRA          622     A
GRA                SB4    X4          (B4) = [LUV]                                    GRA          623     A
GRA                S"TB"  X5+B3       TB = O.SEQ + 3                                  GRA          624     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SA5    X5          R1 = O.SEQ                                  GRA        625    A
GRA                SA0    X3          (A0) = [O.UDT]                              GRA        626    A
GRA                                                                               GRA        627    A
GRA                PROCESS (BOS,EOS)                                              GRA        628    A
GRA                SX7    X5                                                      GRA        629    A
GRA                ZR     X7,ERC1     IF IH[R1] = 0 */ NO BLOCK NUMBER            GRA        630    A
GRA                LX5    -R1.H2P                                                 GRA        631    A
GRA                MX4    -CA.APLNL                                               GRA        632    A
GRA                BX6    -X4*X5                                                  GRA        633    A
GRA                LX6    CA.APLNP    APF = APLN[R1]                              GRA        634    A
GRA                SA6    APF                                                     GRA        635    A
GRA                                                                               GRA        636    A
GRA       *        CLEAR LINK WORD ( NON CANIDATE )                               GRA        637    A
GRA                                                                               GRA        638    A
GRA                PROCESS CLR                                                    GRA        639    A
GRA        ERC1    MX7    0                                                       GRA        640    A
GRA                SA7    A5+B3       LI = R1 + 3;  [LI] = 0  */ CLEAR LINK WORD  GRA        641    A
GRA                                                                               GRA        642    A
GRA       *        ADVANCE TO NEXT INSTRUCTION                                    GRA        643    A
GRA                                                                               GRA        644    A
GRA        ERC2    SA5    A5+4        R1 = R1 + 4                                 GRA        645    A
GRA                UX6    B2,X5                                                   GRA        646    A
GRA                JP     ERC.JT+B2   JUMP( ERC.JT( OC[R1] ) )                    GRA        647    A
GRA        SCON    SPACE  3,14                                                    GRA        648    A
GRA       *        CONSTANT CANIDATE PROCESSING                                   GRA        649    A
GRA                                                                               GRA        650    A
GRA                PROCESS S                                                      GRA        651    A
GRA                LX5    -R1.INP                                                 GRA        652    A
GRA                SX6    X5                                                      GRA        653    A
GRA                AX6    59                                                      GRA        654    A
GRA                SX7    X5                                                      GRA        655    A
GRA                BX6    X6-X7                                                   GRA        656    A
GRA                LX6    IH.CAP      C2W = IHW(0,0,ABS(IN[R1]),0)                GRA        657    A
GRA                                                                               GRA        658    A
GRA        SET1    SA3    A5+2        DI = R1 + 2                                 GRA        659    A
GRA                LX3    59-D.PRSP                                               GRA        660    A
GRA                PL     X3,SET2     IF ^PRS[DI]                                 GRA        661    A
GRA                SA2    A5+4        R1RS = R1 + 4                               GRA        662    A
GRA                LX2    59-R1.SOP-SO.INVP                                       GRA        663    A
GRA                PL     X2,ERC1     IF ^INV[R1RS]  */ CHANGED IN INNER LOOP     GRA        664    A
GRA                SX7    X5                                                      FCC0513      4    A
GRA                PL     X7,SET2     IF IN[R1]  \ 0  */ *S* WITH NEGATIVE CON    FCC0513      5    A
GRA                MX7    -IH.CAL                                                 FCC0513      6    A
GRA                BX6    -X7*X5                                                  FCC0513      7    A
GRA                LX6    IH.CAP      C2W = IHW(0,0,IN[R1],0)                     FCC0513      8    A
GRA                                                                               GRA        665    A
GRA        SET2    MX4    0           R = 0                                       GRA        666    A
GRA                SX1    B1                                                      GRA        667    A
GRA                MX7    0           CCW = 0                                     GRA        668    A
GRA                LX1    CA.CONP     CPW = CAW(0,CON,0)                          GRA        669    A
GRA                MX2    1                                                       GRA        670    A
GRA                LX2    1+IH.MSKP         */ FORCE *S* TO BE A B-CANIDATE       GRA        671    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                BX2    -X7*X2                                                GRA          672          I
          -FCC0513
GRA                BX2    -X6*X2                                                FCC0513        9     A
GRA                LX2    CA.RFP-IH.MSKP                                        GRA          673     A
GRA                BX1    X2+X1       RF[CPW] = ^MSK[C2W]                       GRA          674     A
GRA                RJ     SCT         ENTER CONSTANT                            GRA          675     A
GRA                EQ     ERC2                                                  GRA          676     A
GRA       FMA      SPACE  2                                                     GRA          677     A
GRA                PROCESS FMA                                                  GRA          678     A
GRA                LX5    -R1.INP                                               GRA          679     A
GRA                MX7    1                                                     GRA          680     A
GRA                LX7    1+IH.MSKP                                             GRA          681     A
GRA                SX4    X5                                                    GRA          682     A
GRA                LX4    IH.CAP                                                GRA          683     A
GRA                BX6    X4+X7       C2W = IHW(1,0,IN[R1],0)                   GRA          684     A
GRA                EQ     SET1                                                  GRA          685     A
GRA       SXT      SPACE  3                                                     GRA          686     A
GRA      *         SXT - ENTER *CA* OF FOLLOWING CON SHIFT                      GRA          687     A
GRA                                                                             GRA          688     A
GRA                PROCESS SXT                                                  GRA          689     A
GRA                SA4    A5+5        R2S = R1 + 5                              GRA          690     A
GRA                LX4    -IH.CAP                                               GRA          691     A
GRA                SX6    X4                                                    GRA          692     A
GRA                LX6    IH.CAP      C2W = IHW(0,0,CA[R2S],0)                  GRA          693     A
GRA                MX4    0           R = 0                                     GRA          694     A
GRA                SX1    B1                                                    GRA          695     A
GRA                MX7    0           CCW = 0                                   GRA          696     A
GRA                LX1    CA.CONP     CPW = CAW(0,CON,0)                        GRA          697     A
GRA                RJ     SCT         ENTER CON                                 GRA          698     A
GRA                                                                             GRA          699     A
GRA      *         CHECK PRED OF *SXT* AND BUMP USES OF CON, IF PRED IS MULTI USE GRA         700     A
GRA                                                                             GRA          701     A
GRA                SA5    A5                                                    GRA          702     A
GRA                SB2    "TB"-B1                                               GRA          703     A
GRA                MX3    -D.USESL+1                                            GRA          704     A
GRA                LX5    -R1.RJP                                               GRA          705     A
GRA                BX7    "RN"X5      R = RJ[R1]                                GRA          706     A
GRA                SA4    B2+X7       DIP = TB-1 + R                            GRA          707     A
GRA                LX4    -D.USESP-1                                            GRA          708     A
GRA                BX7    -X3*X4                                                GRA          709     A
GRA                ZR     X7,ERC2     IF USES[DIP] = 1                          GRA          710     A
GRA                SX3    B1                                                    GRA          711     A
GRA                LX3    CA.NOCCP                                              GRA          712     A
GRA                IX6    X3+X6       NOCC[CAW] = NOCC[CAW] + 1                 GRA          713     A
GRA                SA6    A6                                                    GRA          714     A
GRA                EQ     ERC2                                                  GRA          715     A
GRA       STT      SPACE  3,14                                                  GRA          716     A
GRA      *         STT - ENTER CONSTANT OR ADDRESS CANIDATE                     GRA          717     A
GRA                                                                             GRA          718     A
GRA                PROCESS STT                                                  GRA          719     A
GRA                SA1    A5+2        DI = R1 + 2                               GRA          720     A
GRA                SA3    A5+B1       R2 = R1 + 1                               GRA          721     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                LX1    59-D.INCP                                          GRA          722     A
GRA                LX3    -IH.RFP                                            GRA          723     A
GRA                BX4    "RN"X3       R = RF[R2]                            GRA          724     A
GRA                PL     X1,STT2      IF ^INC[DI]   */ NOT AN INCREMENT     GRA          725     A
GRA                                                                          GRA          726     A
GRA                SB2    "TB"-B1                                            GRA          727           I
          -FCC1039
GRA                SA2    B2+X4        DJ = TB-1 + R                         GRA          728           I
          -FCC1039
GRA                LX2    59-D.LDP                                           GRA          729           I
          -FCC1039
GRA                PL     X2,ERC1      IF ^LD[DJ]    */ THIS IS THE TEST INCR GRA          730           I
          -FCC1039
GRA                LX1    D.INCP-D.PSP                                       FCC1039        7     A
GRA                PL     X1,ERC1      IF ^PS[DI]    */ THIS IS THE TEST INCR FCC1039        8     A
GRA                                                                          GRA          731     A
GRA                LX3    IH.RFP-IH.CAP                                      GRA          732     A
GRA                SX6    X3                                                 GRA          733     A
GRA                PL     X6,STT1                                            GRA          734     A
GRA                BX6    -X6                                                GRA          735     A
GRA       STT1     LX6    IH.CAP       C2W = IHW(0,0,ABS(CA[R2]),0)          GRA          736     A
GRA                EQ     SET2                                               GRA          737     A
GRA                                                                          GRA          738     A
GRA      *         PROCESS ADDRESS CANIDATE                                  GRA          739     A
GRA                                                                          GRA          740     A
GRA       STT2     BX6    X0*X3        C2W = IHW(0,0,CAIH[R2])               GRA          741     A
GRA                LX5    -R1.INP                                            GRA          742     A
GRA                SX1    B1                                                 GRA          743     A
GRA                LX6    IH.RFP                                             GRA          744     A
GRA                MX7    -R1.H2L                                            GRA          745     A
GRA                LX1    CA.ACP                                             GRA          746     A
GRA                SX2    X5                                                 GRA          747     A
GRA                LX5    R1.INP-R1.H2P                                      GRA          748     A
GRA                BX1    X1+X2        CPW = CAW(0,AC,0,IN[R1])              GRA          749     A
GRA                BX7    -X7*X5                                             GRA          750     A
GRA                LX7    CC.H2P       CCW = H2[R1]                          GRA          751     A
GRA                EQ     TLD2                                               GRA          752           I
          -FCC0919
GRA                EQ     ST2A                                               FCC0919        5     A
GRA       TST      SPACE  3,14                                               GRA          753     A
GRA      *         TST/TLD - ENTER VALUE CANIDATES                           GRA          754     A
GRA                                                                          GRA          755     A
GRA                PROCESS TST                                               GRA          756     A
GRA                SA3    A5+B3        LI = R1 + 3                           GRA          757     A
GRA                ZR     X3,ERC2      IF [LI] = 0   */ USED IN INNER LOOP   GRA          758     A
GRA                SA1    TSTA                                               GRA          759     A
GRA                EQ     TLD1                                               GRA          760     A
GRA                                                                          GRA          761     A
GRA                PROCESS TLD                                               GRA          762     A
GRA                SA1    TLDA                                               GRA          763     A
GRA                                                                          GRA          764     A
GRA       TLD1     SA3    A5+B1        R2 = R1 + 1                           GRA          765     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                MX7    0           CCW = 0                                    GRA        766     A
GRA                SX4    B0          R = 0                                      GRA        767     A
GRA                BX6    X3          C2W = [R2]                                 GRA        768     A
GRA                                                                              GRA        769     A
GRA       TLD2     RJ     SCT         ENTER CANIDATE                             GRA        770         I
         -CCG16
GRA       TLD2     SX3    B2-OC.LDC                                              CCG16      206     A
GRA                MI     X3,TLD3     IF OC[R1] < OC.LDC  */ ^(LDC ! LDV)        CCG16      207     A
GRA                SX3    X3+B1                                                  CCG16      208     A
GRA                LX3    IH.SLVP     SLV[C2W] = 1 + OC[R1]-OC.LDC               CCG16      209     A
GRA                BX6    X3+X6                                                  CCG16      210     A
GRA                                                                              CCG16      211     A
GRA       TLD3     RJ     SCT         ENTER CANIDATE                             CCG16      212     A
GRA                EQ     ERC2                                                   GRA        771     A
GRA                                                                              GRA        772     A
GRA       TLDA     BFMW   CA,(PL,USE)                                            GRA        773     A
GRA       TSTA     BFMW   CA,(PL,DEF)                                            GRA        774     A
GRA       LDST     SPACE  2,14                                                   GRA        775     A
GRA      *         LD/ST - ENTER VALUE OR ADDRESS CANIDATE                       GRA        776     A
GRA                                                                              GRA        777     A
GRA                PROCESS ST                                                    GRA        778     A
GRA                SX1    B1                                                     GRA        779     A
GRA                LX1    CA.DEFP     BIT = DEF                                  GRA        780     A
GRA                EQ     ST1                                                    GRA        781     A
GRA                                                                              GRA        782     A
GRA                PROCESS (LD,ILD)                                              GRA        783         I
         -CCG1
GRA                PROCESS (LD,ILD,LDC)                                          CCG1        92         I
         -CCG16
GRA                PROCESS (LD,ILD,LDC,LDV)                                      CCG16      213     A
GRA                SA4    A5+2        DI = R1 + 2                                GRA        784     A
GRA                SX1    B1                                                     GRA        785     A
GRA                LX1    D.RFP                                                  GRA        786     A
GRA                BX2    X1*X4       RFB = RF[DI]  */ FORCE *RF* IF *DO* LIMITS GRA        787     A
GRA                LX1    CA.USEP-D.RFP                                          GRA        788     A
GRA                LX2    CA.RFP-D.RFP                                           GRA        789     A
GRA                BX1    X1+X2       BIT = (USE,RFB)                            GRA        790     A
GRA                                                                              GRA        791     A
GRA       ST1      SA3    A5+B3       LI = R1 + 3                                GRA        792     A
GRA                LX5    -R1.INP                                                GRA        793     A
GRA                SX6    X5                                                     GRA        794     A
GRA                BX1    X1+X6       CPW = CAW(0,BIT,IN[R1])                    GRA        795     A
GRA                SA2    A5+B1       R2 = R1 + 1                                GRA        796     A
GRA                LX3    59-CA.VCP                                              GRA        797     A
GRA                MX7    0           CCW = 0                                    GRA        798     A
GRA                SX4    B0          R = 0                                      GRA        799     A
GRA                BX6    X2          C2W = [R2]                                 GRA        800     A
GRA                MI     X3,TLD2     IF VC[LI]                                  GRA        801     A
GRA                LX3    CA.VCP-CA.ACP                                          GRA        802     A
GRA                PL     X3,ST4      IF ^AC[LI]                                 GRA        803     A
GRA                                                                              GRA        804     A
GRA      *         ADJUST RF FOR *SCT* IF IT IS AN *INC*                         GRA        805     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                                                                    GRA        806    A
GRA              LX2    -IH.RFP                                        GRA        807    A
GRA              SB2    "TB"-B1                                        GRA        808         I
         -FCC0919
GRA              BX4    "RN"X2      R = RF[R2]                         GRA        809         I
         -FCC0919
GRA              BX4    "RN"X2      R = RF[R2]                         FCC0919      6    A
GRA                                                                    FCC0919      7    A
GRA       ST2    SB2    "TB"-B1                                        FCC0919      8    A
GRA              SA2    B2+X4       DIP = TB-1 + R                     GRA        810         I
         -FCC1039
GRA              MX5    -IH.CAIHL                                      GRA        811    A
GRA              BX6    -X5*X6      C2W = IHW(0,0,CAIH[R2])            GRA        812    A
GRA                                                                    FCC1039      9    A
GRA       ST2B   SA2    B2+X4       DIP = TB-1 + R                     FCC1039     10    A
GRA              LX2    59-D.INCP                                      GRA        813    A
GRA              PL     X2,ST3      IF ^INC[DIP]                       GRA        814    A
GRA                                                                    GRA        815    A
GRA              LX2    1+D.INCP+58-D.TYP                              GRA        816    A
GRA              SA4    A2-B1       R2P = DIP - 1                      GRA        817    A
GRA              PL     X2,ST2      IF TYPE[DIP] = I                   GRA        818    A
GRA              LX4    -IH.RFP                                        GRA        819    A
GRA              BX4    "RN"X4      R = RF[R2P]   */ POINT TO INC VARIABLE GRA    820    A
GRA              EQ     TLD2                                           GRA        821         I
         -FCC1039
GRA              EQ     ST2B                                           FCC1039     11    A
GRA                                                                    GRA        822    A
GRA       ST2    SA2    A4-B1       R1P = R2P - 1                      GRA        823         I
         -FCC1039
GRA              LX2    -R1.RJP                                        GRA        824         I
         -FCC1039
GRA              BX4    "RN"X2      R = RJ[R1]                         GRA        825         I
         -FCC1039
GRA              SA3    "TB"+X4     LIP = TB + R                       GRA        826         I
         -FCC1039
GRA              SA3    B5+X3       CI = RCT([LIP])                    GRA        827         I
         -FCC1039
GRA              LX3    59-CA.DEFP                                     GRA        828         I
         -FCC1039
GRA              MI     X3,TLD2     IF DEF[CI]    */ MUST BE THE *INC* VARIABLE GRA 829    I
         -FCC1039
GRA              LX2    R1.RJP-R1.RKP                                  GRA        830         I
         -FCC1039
GRA              BX4    "RN"X2      R = RK[R2]                         GRA        831         I
         -FCC1039
GRA              EQ     TLD2                                           GRA        832         I
         -FCC1039
GRA       ST2    SA2    A2+B1       LIP = DIP + 1 */ CONTAINS CANIDATE ORDINAL FCC1039  12    A
GRA              ZR     X2,ST3      IF [LI] = 0   */ THIS INC IS NOT A CANIDATE FCC1039 13    A
GRA              SX2    X2                                             FCC1039     14    A
GRA              MX4    0           R = 0                              FCC1039     15    A
GRA              LX2    IH.RFP                                         FCC1039     16    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                BX6    X2+X6       RF[C2W] = CO[LIP]                          FCC1039      17     A
GRA                                                                              GRA         833     A
GRA        ST3     RJ     SCT         ENTER CANIDATE                             GRA         834     A
GRA                SA4    A5+3                                                   GRA         835     A
GRA                NZ     X4,ERC2     IF [R1+3] " 0 */ CANIDATE ENTERED          GRA         836     A
GRA                                                                              GRA         837     A
GRA                SA1    A5+                                                    GRA         838     A
GRA                LX1    -R1.INP                                                GRA         839     A
GRA        ST      SPACE  1,12                                                   GRA         840     A
GRA       *        SET BIT IN *LUV* TO INDICATE USEAGE OF VAR IN AN INNER LOOP   GRA         841     A
GRA                                                                              GRA         842     A
GRA        ST4     SB2    X1+B1                                                  GRA         843     A
GRA                SA2    A0+B2       U2 = UDT(IN[R1]+1)                         GRA         844     A
GRA                SX1    B1                                                     GRA         845     A
GRA                SA3    B4+X2       LUA = LUV + WI[U2]                         GRA         846     A
GRA                UX7    B2,X2                                                  GRA         847     A
GRA                LX1    B2,X1                                                  GRA         848     A
GRA                BX6    X1+X3       [LUA] = [LUA] ! SHIFT(1,BITN[U2])          GRA         849     A
GRA                SA6    A3                                                     GRA         850     A
GRA                EQ     ERC1                                                   GRA         851     A
GRA        RS      SPACE  3,20                                                   GRA         852     A
GRA       **       RS - IF A FULL LOCK *RS* AND ONLY 1 INNER LOOP IN THE NEST,   GRA         853     A
GRA       *        THEN ATTEMPT TO MOVE IT OUT.                                  GRA         854     A
GRA                                                                              GRA         855     A
GRA                PROCESS RS                                                    GRA         856     A
GRA                SA1    N.HB                                                   GRA         857     A
GRA                LX5    -R1.SOP-SO.LKP                                         GRA         858     A
GRA                MX6    -SO.LKL                                                GRA         859     A
GRA                SB2    X1                                                     GRA         860     A
GRA                NE     B2,B1,RS9   IF N.HB " 1   */ ^ OUTER WITH 1 INNER LOOP GRA         861     A
GRA                BX7    -X6*X5                                                 GRA         862     A
GRA                SB2    X7+B1                                                  GRA         863     A
GRA                NE     B2,B3,RS9   IF SOLK[R1] " 2 */ NOT FULL LOCK RS        GRA         864     A
GRA                MX6    -SO.REGL                                               GRA         865     A
GRA                SA2    XRF                                                    GRA         866     A
GRA                NZ     X2,RS9      IF XRF " 0    */ EXT REFS IN LOOP          GRA         867     A
GRA                LX5    SO.LKP                                                 GRA         868     A
GRA                BX7    -X6*X5      REGN = SOREG[R1]                           GRA         869     A
GRA                                                                              GRA         870     A
GRA       *        BYPASS MOVING ASSIGNMENT OUT OF THEIR IS A TEMP/RJRS USE      GRA         871     A
GRA                                                                              GRA         872     A
GRA                SA2    TRA                                                    GRA         873     A
GRA                SB2    X7-59                                                  GRA         874     A
GRA                LX6    -B2,X2                                                 GRA         875     A
GRA                MI     X6,RS9      IF SHIFT(TRA,59-REGN) < 0  */ TEMP/RJRS USE GRA         876     A
GRA                                                                              GRA         877     A
GRA       *        FIND PRED OF THE *RS* THAT IS IN *RCT*                        GRA         878         I
          -FCC0448
GRA                                                                              GRA         879         I
          -FCC0448
GRA                SA4    A5-B1       LI = R1 - 1                                GRA         880         I
          -FCC0448

          0        1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                 SA3    A4-B1       DI = LI - 1                                  GRA          881          I
          -FCC0448
GRA                 NZ     X4,RS1      IF [LI] " 0   */ PRED IN *RCT*               GRA          882          I
          -FCC0448
GRA                                                                                 GRA          883          I
          -FCC0448
GRA                 LX3    58-D.TYP                                                 GRA          884          I
          -FCC0448
GRA                 MX6    D.TYL                                                    GRA          885          I
          -FCC0448
GRA                 BX3    X6*X3                                                    GRA          886          I
          -FCC0448
GRA                 ZR     X3,RS0      IF TYPE[DI] = I                              GRA          887          I
          -FCC0448
GRA                 EQ     ERC1                                                     GRA          888          I
          -FCC0448
GRA                                                                                 GRA          889          I
          -FCC0448
GRA        RS0      SA3    A3-2        R1P = DI - 2                                 GRA          890          I
          -FCC0448
GRA                 UX6    B2,X3                                                    GRA          891          I
          -FCC0448
GRA                 LX3    -R1.RJP                                                  GRA          892          I
          -FCC0448
GRA                 BX1    "RN"X3      R = RJ[R1P]                                  GRA          893          I
          -FCC0448
GRA                 SA4    "TB"+X1     LI = TB + R                                  GRA          894          I
          -FCC0448
GRA                 ZR     X4,ERC1     IF [LI] = 0   */ PRED NOT IN *RCT*           GRA          895          I
          -FCC0448
GRA                 SX6    B2-OC.SA                                                 GRA          896          I
          -FCC0448
GRA                 ZR     X6,RS1      IF OC[R1P] = OC.SA                           GRA          897          I
          -FCC0448
GRA                 SX6    B2-OC.XMT                                                GRA          898          I
          -FCC0448
GRA                 NZ     X6,ERC1     IF OC[R1P] " OC.XMT                          GRA          899          I
          -FCC0448
GRA                                                                                 GRA          900          I
          -FCC0448
GRA        RS1      SA3    B5+X4       CAW = RCT([LI])                              GRA          901          I
          -FCC0448
GRA                 RJ     FCP         I = FCP(R1)   */ RCT ORD OF PRED             FCC0448       69     A
GRA                 ZR     X4,ERC1     IF I = 0      */ PRED NOT A CANIDATE         FCC0448       70     A
GRA                                                                                 FCC0448       71     A
GRA                 SA3    B5+X4       CAW = RCT(I)                                 FCC0448       72     A
GRA                 LX3    59-CA.RAP                                                GRA          902     A
GRA                 MI     X3,RS9      IF RA[CAW]    */ CANIDATE ASSIGNED           GRA          903     A
GRA        RS       SPACE  2,14                                                     GRA          904     A
GRA       *         MOVE THE *RS* AND ITS PRED IF THE *CAIH* IN THE *RS* MATCHS     GRA          905     A
GRA       *         THE *CAIH* IN THE *LD* OR *RS* IS INVARIANT IN BOTH LOOPS.      GRA          906     A
GRA                                                                                 GRA          907     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                 SA3    A5+B1       R2 = R1 + 1;  R2W = [R2]                          GRA          908     A
GRA                 LX5    -SO.INVP                                                      GRA          909     A
GRA                 ZR     X3,RS2      IF R2W = 0    */ NO *CAIH* IN *RS*                GRA          910     A
GRA                 SA2    A4-2        R2P = LI - 2                                      GRA          911     A
GRA                 BX6    X2-X3                                                         GRA          912     A
GRA                 NZ     X6,RS2      IF[R2P] = R2W                                     GRA          913          I
          -FCC0414
GRA                 NZ     X6,RS2      IF [R2P] " R2W                                    FCC0414        4          I
          -FCC1770X
GRA                                                                                      GRA          914          I
          -FCC1770X
GRA                 SA2    B5+X4       CI = RCT([LI])                                    GRA          915          I
          -FCC1770X
GRA                 LX2    59-CA.VCP                                                     GRA          916          I
          -FCC1770X
GRA                 SA2    B5+X4       CI = RCT([LI])                                    FCC1770X       6     A
GRA                 ZR     X6,RS1      IF [R2P] = R2W  */ SAME IH"S                      FCC1770X       7     A
GRA                                                                                      FCC1770X       8     A
GRA       *         DO NOT MOVE *RS* PREDECESSOR IS  LD FP AND *AC*                      FCC1770X       9     A
GRA                                                                                      FCC1770X      10     A
GRA                 LX2    59-CA.FPP                                                     FCC1770X      11     A
GRA                 PL     X2,RS2      IF ^FP[CI]                                        FCC1770X      12     A
GRA                 LX2    CA.FPP-CA.ACP                                                 FCC1770X      13     A
GRA                 PL     X2,RS2      IF ^AC[CI]                                        FCC1770X      14     A
GRA                 EQ     ERC1                                                          FCC1770X      15     A
GRA                                                                                      FCC1770X      16     A
GRA        RS1      LX2    59-CA.VCP                                                     FCC1770X      17     A
GRA                 MI     X2,RS3      IF VC[CI]     */ NO INTERFERENCE IN LOOPS         GRA          917     A
GRA                 EQ     ERC1                                                          GRA          918     A
GRA                                                                                      GRA          919     A
GRA        RS2      MX6    -2                                                            GRA          920     A
GRA                 BX2    X6+X5                                                         GRA          921     A
GRA                 NZ     X2,RS9      IF ^( INV[R1] & INVC[R1] )                        GRA          922     A
GRA                                                                                      GRA          923     A
GRA                 SA1    A5-4        R1P = R1 - 4                                      GRA          924     A
GRA                 SA2    A1+B1       R2P = R1P + 1                                     GRA          925     A
GRA                 UX6    B2,X1                                                         GRA          926     A
GRA                 SX3    B2-OC.STT                                                     GRA          927     A
GRA                 NZ     X3,RS3      IF OC[R1P] " OC.STT                               GRA          928     A
GRA                 SX6    X2                                                  002180    FCC2074        5     A
GRA                 LX2    -IH.RFP                                                       GRA          929     A
GRA                 BX1    "RN"X2      R = RF[R2P]                                       GRA          930     A
GRA                 ZR     X1,RS3      IF R = 0      */ NO PREDECESSOR                   GRA          931     A
GRA                                                                                      GRA          932     A
GRA                 SA2    "TB"+X1     LIP = TB + R;  K = CO[LI]                         GRA          933     A
GRA                 SA3    A5-B1       LI = R1 - 1;  J = CO[LI]                          GRA          934     A
GRA                 ZR     X2,ERC1     IF K = 0      */ PRED NOT IN *RCT*                GRA          935          I
          -FCC2074
GRA                 LX3    RA.R3P                                                        GRA          936          I
          -FCC2074
GRA                 SA1    B5+X2       CK = RCT(K)                                 002200    FCC2074        6     A
GRA                 ZR     X6,RS2A     IF IH[R2P] = 0  */ NO IH                    002210    FCC2074        7     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                BX6    X1                                                  002220   FCC2074      8    A
GRA                LX6    59-CA.VCP                                           002230   FCC2074      9    A
GRA                PL     X6,ERC1     IF ^VC[CK]    */ FP AND INTERFERENCE    002240   FCC2074     10    A
GRA                                                                           002250   FCC2074     11    A
GRA       RS2A     LX3    RA.R3P                                              002260   FCC2074     12    A
GRA                BX6    X2+X3       (R1,R3)RAT(REGN) = (J,K)                         GRA        937         I
          -FCC1014
GRA                BX6    X2+X3       (R1,R3)RAT(REGN) = (K,J)                         FCC1014     35    A
GRA                SA6    RAT+X7                                                       GRA        938    A
GRA                SX3    B1                                                           GRA        939    A
GRA                SA1    B5+X2             CK = RCT(K)                                GRA        940         I
          -FCC2074
GRA                LX3    CA.SUSEP                                                     GRA        941    A
GRA                IX6    X1-X3       SUSE[CK] = SUSE[CK] - 1                          GRA        942    A
GRA                SA6    A1                                                           GRA        943    A
GRA       .FTN     IFEQ   HC.ID,2                                   ?FTN              CCG4        67    A
GRA                SB2    X1-2                                                         FCC1014     36    A
GRA                NZ     B2,RS3      IF UDI[CK] " 1                                   FCC1014     37    A
GRA                LX1    59-CA.VCP                                                    FCC1014     38    A
GRA                PL     X1,RS3      IF ^VC[CK]                                       FCC1014     39    A
GRA                SA1    A1+B1       C2K = RCT(K+1)                                   FCC1014     40    A
GRA                LX1    -IH.RFP                                                      FCC1014     41    A
GRA                SB2    X1                                                           FCC1014     42    A
GRA                NE     B2,B1,RS3   IF RF[C2K] " 1  */ ^ A0                          FCC1014     43    A
GRA                                                                                   FCC1014     44    A
GRA       *        STT  RI,,FPIH  EXPANDED BY *ESR*, CHANGE IT BACK TO NON              FCC1014     45    A
GRA       *        EXPANDED FORM IN *RCT* SO WE MAY USE IT AS A BASE ADDRESS            FCC1014     46    A
GRA       *        FOR OTHER CANDIDATES.                                                FCC1014     47    A
GRA                                                                                   FCC1014     48    A
GRA                SA3    A5-B1       J = CO[LI]                                       FCC1014     49    A
GRA                LX1    IH.RFP-IH.CAP                                                FCC1014     50    A
GRA                SX1    X1+2        IH = CA[C2K] + 2                                 FCC1014     51         I
          -CCG4
GRA                SB2    X1                                                           CCG4        68         I
          -CCG18
GRA                SA2    =XO.FPT                                                      CCG4        69         I
          -CCG18
GRA                SA1    B2+X2       IH = FPT(CA[C2K])                                CCG4        70         I
          -CCG18
GRA                SB2    X1                                                           CCG18      107    A
GRA                SA2    =XO$FPI                                                      CCG18      108    A
GRA                SA1    X2+B2                                                        CCG18      109    A
GRA                LX1    -FP.PNTP    IH = PNT[FPI(CA[C2K])]                           CCG18      110    A
GRA                SX1    X1                                                           CCG18      111    A
GRA                SB2    X3+B1                                                        FCC1014     52    A
GRA                SA3    B5+B2       C2J = RCT(J+1)                                   FCC1014     53    A
GRA                BX6    X1+X3                                                        FCC1014     54    A
GRA                MX1    -IH.RFP                                                      FCC1014     55    A
GRA                BX6    -X1*X6      RCT(J+1) = IHW(0,CA[C2J],IH)                     FCC1014     56    A
GRA                SA6    A3                                                           FCC1014     57    A
GRA                MX6    0                                                            FCC1014     58    A
GRA                SA6    RAT+X7      RAT(REGN) = 0                                    FCC1014     59    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA        .FTN     ENDIF                                                 'FTN              CCG4          71     A
GRA        RS       SPACE  1,10                                                             GRA          944     A
GRA       *         SETUP *RCT* AND *RAT* TO MARK THE CANIDATE AS ASSIGNED                  GRA          945     A
GRA       *         TO A REGISTER IN THIS LOOP.                                             GRA          946     A
GRA                                                                                         GRA          947     A
GRA        RS3      SA2    B5+X4       J = CO[LI];  CI = RCT(J)                             GRA          948     A
GRA                 SA3    A2+2                                                             GRA          949     A
GRA                 MX1    2                                                                GRA          950     A
GRA                 LX1    2+CA.RAFP   (RA,IRA)RCT(J) = 1                                   GRA          951     A
GRA                 BX6    X1+X2                                                            GRA          952     A
GRA                 SA6    A2                                                               GRA          953     A
GRA                 SB2    X7-8                                                             GRA          954     A
GRA                 LX7    CC.REG1P                                                         GRA          955     A
GRA                 BX6    X3+X7       REG1[RCT(J+2)] = REGN                                GRA          956     A
GRA                 SX1    2S6                                                              GRA          957           I
          -FCC1014
GRA                 BX6    X1+X6       MIT[RCT(J+2)] = 2  */ IN CASE J IS AN *AC*           GRA          958           I
          -FCC1014
GRA                 LX2    -CA.ACP                                                          FCC1014       60     A
GRA                 SX1    B1                                                               FCC1014       61     A
GRA                 BX1    X1*X2                                                            FCC1014       62     A
GRA                 LX1    CC.MITP+1                                                        FCC1014       63     A
GRA                 BX6    X1+X6       MIT[RCT(J+2)] = 2*AC[CI]                             FCC1014       64     A
GRA                 SA6    A3                                                               GRA          959     A
GRA                 LX7    -CC.REG1P                                                        GRA          960     A
GRA                 SX6    X4                                                               GRA          961     A
GRA                 SA6    A5+3        [R1+3] = J                                           GRA          962     A
GRA                 SA1    RAT+X7                                                           GRA          963     A
GRA                 NZ     X1,RS4      IF RAT(REGN) " 0  */ STT CASE                        GRA          964     A
GRA                 SA6    A1+         RAT(REGN) = J                                        GRA          965     A
GRA                                                                                         GRA          966     A
GRA        RS4      PL     B2,RS5      IF REGN > 7                                          GRA          967     A
GRA                                                                                         GRA          968     A
GRA                 SA7    FBA         FBA = REGN    */ INIDCATE FINAL B-ASSIGNMEN          GRA          969     A
GRA                 EQ     ERC2                                                             GRA          970     A
GRA                                                                                         GRA          971     A
GRA        RS5      SA1    NXC                                                              GRA          972     A
GRA                 SA6    XCT+X1      XCT(NXC) = J                                         GRA          973     A
GRA                 SX7    X1+B1                                                            GRA          974     A
GRA                 SA7    A1          NXC = NXC + 1                                        GRA          975     A
GRA                 EQ     ERC2                                                             GRA          976     A
GRA                                                                                         GRA          977     A
GRA       *         CANNOT MOVE ASSIGNMENT, MARK PRED AS *EU*                               GRA          978           I
          -FCC2056
GRA                                                                                         GRA          979           I
          -FCC2056
GRA        RS9      SA4    A5-B1       LI = R1 - 1;  I = CO[LI]                             GRA          980           I
          -FCC0448  -FCC2056
GRA        RS9      RJ     FCP         I = FCP(I)                                           FCC0448       73           I
          -FCC2056
GRA                 ZR     X4,ERC1     IF I = 0                                             GRA          981           I
          -FCC2056

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SA2    B5+X4                                                GRA          982          I
          -FCC2056
GRA                SX1    B1                                                   GRA          983          I
          -FCC2056
GRA                LX1    CA.EUP                                               GRA          984          I
          -FCC2056
GRA                BX6    X1+X2       EU[RCT(I)] = 1                           GRA          985          I
          -FCC2056
GRA                SA6    A2                                                   GRA          986          I
          -FCC2056
GRA                EQ     ERC1                                                 GRA          987          I
          -FCC2056
GRA        RS9     EQU    ERC1        MARKED AS *EU* IN DUM                    FCC2056        6     A
GRA        FCP     SPACE  3,14                                                 FCC0448       74     A
GRA       **       FCP - FIND CANIDATE PREDECESSOR                             FCC0448       75     A
GRA       *                                                                    FCC0448       76     A
GRA       *        ENTRY  (A5) = R1 , OF AN *RS* INSTRUCTION                   FCC0448       77     A
GRA       *                                                                    FCC0448       78     A
GRA       *        EXIT   (X4) = I , RCT ORDINAL OF CANIDATE                   FCC0448       79     A
GRA                                                                            FCC0448       80     A
GRA        FCP     ROUTINE                                                     FCC0448       81     A
GRA                SA4    A5-B1       LI = R1 - 1;  I = [I]                    FCC0448       82     A
GRA                SA3    A4-B1       DI = LI - 1                              FCC0448       83     A
GRA                NZ     X4,FCP      IF I " 0      */ IMMEDIATE PRED A CANIDATE FCC0448       84     A
GRA                LX3    58-D.TYP                                             FCC0448       85     A
GRA                MX6    D.TYL                                                FCC0448       86     A
GRA                BX3    X6*X3                                                FCC0448       87     A
GRA                NZ     X3,FCP      IF TYPE[DI] " I  */ NOT POSSIBLE SA/XMT  FCC0448       88     A
GRA                                                                            FCC0448       89     A
GRA                SA3    A3-2        R1P = DI - 2                             FCC0448       90     A
GRA                UX6    B2,X3                                                FCC0448       91     A
GRA                LX3    -R1.RJP                                              FCC0448       92     A
GRA                BX6    "RN"X3      R = RJ[R1P]                              FCC0448       93     A
GRA                SA4    "TB"+X6     LI = TB + R;  I = [LI]                   FCC0448       94     A
GRA                ZR     X4,FCP      IF I " 0      */ PRED NOT A CANIDATE     FCC0448       95     A
GRA                SX6    B2-OC.SA                                             FCC0448       96     A
GRA                ZR     X6,FCP      IF OC[R1P] = OC.SA                       FCC0448       97     A
GRA                SX6    B2-OC.XMT                                            FCC0448       98     A
GRA                ZR     X6,FCP      IF OC[R1P] = OC.XMT                      FCC0448       99     A
GRA                                                                            FCC0448      100     A
GRA                MX4    0           I = 0         */ INDICATE PRED NOT A CANIDA FCC0448      101     A
GRA                EQ     FCP                                                  FCC0448      102     A
GRA        SCT     TITLE  SCT - SEARCH CANIDATE TABLE, ENTER UPDATE ENTRY      GRA          988     A
GRA       **       SCT - SEARCH CANIDATE TABLE FOR MATCH OF KEY AND MAKE NEW   GRA          989     A
GRA       *        ENTRY OR UPDATE OLD ONE.                                    GRA          990     A
GRA       *                                                                    GRA          991     A
GRA       *        ENTRY  (X1) = CPW , CANIDATE PROPERTY WORD                  GRA          992     A
GRA       *               (X4) = R , RF OF INDEXED REFERENCE                   GRA          993     A
GRA       *               (X6) = C2W , CANIDATE KEY                            GRA          994     A
GRA       *               (X7) = CCW , THIRD WORD OF ENTRY                     GRA          995     A
GRA       *               (A5) = R1 , INSTRUCTION ADDRESS                      GRA          996     A
GRA       *               (B3) = 3                                             GRA          997     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA      *                (B5) = CO , O.RCT                                    GRA        998     A
GRA      *                (B6) = CL , LWA+1 OF TABLE                           GRA        999     A
GRA      *                                                                     GRA       1000     A
GRA      *         EXIT   (A6,X6) = UPDATED *CA* WORD OF ENTRY                 GRA       1001     A
GRA      *                CANIDATE ENTERED, USES, NOCC BUMPED, ETC             GRA       1002     A
GRA      *                                                                     GRA       1003     A
GRA      *         USES   B - 2, A - 1, 2, 3, 4  X - 1, 2, 3, 4, 5, 6, 7       GRA       1004     A
GRA                                                                            GRA       1005     A
GRA       SCT0     SX7    0                                                    GRA       1006     A
GRA                SA7    A5+3        [R1+3] = 0                               GRA       1007     A
GRA                                                                            GRA       1008     A
GRA       SCT      ROUTINE                                                     GRA       1009     A
GRA                SB2    B5+B3       SA = CO + 3   */ SEARCH ADDRESS          GRA       1010     A
GRA                MX2    -CC.H2L                                              GRA       1011          I
          -FCC1014
GRA                ZR     X4,SCT1     IF R = 0                                 GRA       1012     A
GRA                SA4    "TB"+X4     LI = TB + R;  R = [LI]  */ SUBST RCT INDX GRA       1013     A
GRA                ZR     X4,SCT0     IF R = 0      */ OPERAND NOT IN RCT      GRA       1014          I
          -FCC1014
GRA                SA3    B5+X4                                                GRA       1015          I
          -FCC1014
GRA                LX3    59-CA.VCP                                            GRA       1016          I
          -FCC1014
GRA                PL     X3,SCT0     IF ^VC[RCT(R)]                           GRA       1017          I
          -FCC1014
GRA                SB2    B2+X4       SA = SA + R                              GRA       1018          I
          -FCC1014
GRA                ZR     X4,SCT0A    IF R = 0      */ OPERAND NOT IN *RCT*    FCC1014     65     A
GRA                SA3    B5+X4                                                FCC1014     66     A
GRA                LX3    59-CA.VCP                                            FCC1014     67     A
GRA                MI     X3,SCT0B    IF VC[RCT(R)]                            FCC1014     68     A
GRA                MX4    0           R = 0                                    FCC1014     69     A
GRA       SCT0A    SX3    X6+                                                  FCC1014     70     A
GRA                ZR     X3,SCT0     IF IH[C2W] = 0  */ *AC* AND NO *IH*      FCC1014     71     A
GRA                                                                            FCC1014     72     A
GRA                SX3    4S6                                                  FCC1014     73     A
GRA                BX7    X3+X7       MIT[CCW] = 4  */ RF NOT IN RCT CASE      FCC1014     74     A
GRA                                                                            FCC1014     75     A
GRA       SCT0B    SB2    B2+X4       SA = SA + R                              FCC1014     76     A
GRA                LX4    IH.RFP                                               GRA       1019     A
GRA                BX6    X4+X6       RF[R2] = R                               GRA       1020     A
GRA                                                                            GRA       1021     A
GRA      *         SET SEARCH TERMINATOR, SEARCH FOR PREVIOUS OCCURANCES      GRA       1022     A
GRA                                                                            GRA       1023     A
GRA       SCT1     SA6    B6+B1       [CL+1] = C2W                             GRA       1024     A
GRA                SA7    A6+B1       [CL+2] = CCW                             GRA       1025     A
GRA                LX2    CC.H2P                                               GRA       1026          I
          -FCC1014
GRA                SA4    B2+B1       CI = SA + 1;  KEY = [CI]                 GRA       1027          I
          -FCC0594
GRA                SA3    A5+B3       LI = R1 + 3                              FCC0594      6     A
GRA                SA4    B2+1        CI = SA + 1;  KEY = [CI]  */ C2 WORD     FCC0594      7          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

          -FCC1014
GRA                 SA4    B2+B1       CI = SA + 1;  KEY = [CI]  */ C2 WORD          FCC1014      77     A
GRA                 SA2    SCTA                                                      FCC1014      78     A
GRA                 BX5    X7                                                        FCC0594       8     A
GRA                 BX1    X3+X1       CPW = CPW ! [LI]  */ SET CT, USAGE            FCC0594       9     A
GRA                 LX7    X1                                                        FCC0594      10     A
GRA                 SA7    B6+         [CL] = CPW    */ SKEL CA WORD                 FCC0594      11     A
GRA                 BX7    X5                                                        FCC0594      12     A
GRA                                                                                  GRA        1028     A
GRA       SCT2      BX5    X4-X6       DIFF = C2W - KEY                              GRA        1029     A
GRA                 SA4    A4+B3       CI = CI + 3;  KEY = [CI]                      GRA        1030     A
GRA                 NZ     X5,SCT2     IF DIFF " 0                                   GRA        1031     A
GRA                                                                                  GRA        1032     A
GRA                 SA3    A4-2        CJ = CI - 2                                   GRA        1033     A
GRA                 BX4    X3-X7                                                     GRA        1034          I
          -FCC0594
GRA                 BX5    -X2*X4                                                    GRA        1035          I
          -FCC0594
GRA                 NZ     X5,SCT2     IF H2[CJ] " H2[CCW]                           GRA        1036          I
          -FCC0594
GRA                 BX5    X3-X7                                                     FCC0594      13     A
GRA                 BX5    -X2*X5                                                    FCC0594      14          I
          -FCC1014
GRA                 NZ     X5,SCT2     IF H2[CJ] " H2[CCW]                           FCC0594      15          I
          -FCC1014
GRA                 BX5    X2*X5                                                     FCC1014      79     A
GRA                 NZ     X5,SCT2     IF (H2,MIT)[CJ] " (H2,MIT)[CCW]               FCC1014      80     A
GRA                 SA3    A3-2        CJ = CJ - 2   */ CA WORD                      FCC0594      16     A
GRA                 BX5    X3-X1                                                     FCC0594      17     A
GRA                 MX3    -CA.CTL                                                   FCC0594      18     A
GRA                 LX5    -CA.CTP                                                   FCC0594      19     A
GRA                 BX3    -X3*X5                                                    FCC0594      20     A
GRA                 NZ     X3,SCT2     IF CT[CJ] " CT[CPW]  */ NOT SAME TYPE         FCC0594      21     A
GRA                                                                                  GRA        1037     A
GRA       *         CHECK RESULTS OF SEARCH                                          GRA        1038     A
GRA                                                                                  GRA        1039     A
GRA                 SB2    A4-4        EA = CI - 4   */ ENTRY ADDRESS                GRA        1040     A
GRA                 LT     B2,B6,SCT3  IF EA < CL    */ ENTRY IN TABLE               GRA        1041     A
GRA                                                                                  GRA        1042     A
GRA                 SA4    A5+2        DI = R1 + 2                                   GRA        1043     A
GRA                 SX2    B1                                                        GRA        1044     A
GRA                 SB6    B6+B3       CL = CL + 3                                   GRA        1045     A
GRA                 LX2    CA.FPP                                                    GRA        1046     A
GRA                 LX4    CA.FPP-D.FPP                                              GRA        1047     A
GRA                 BX3    X2*X4                                                     GRA        1048     A
GRA                 BX2    X3+X1       CAW = CPW ! FP[DI]                            GRA        1049     A
GRA                 EQ     SCT4                                                      GRA        1050     A
GRA       SCT       SPACE  1,10                                                      GRA        1051     A
GRA       SCT3      SA3    B2+                                                       GRA        1052          I
          -FCC0656
GRA       SCT3      SA3    B2                                                        FCC0656      54     A
GRA                 MX5    -CA.UDIL                                                  FCC0656      55     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                BX1    X5*X1       UDI[CPW] = 0                                FCC0656      56     A
GRA                BX2    X1+X3       CAW = CPW ! [EA]                            GRA        1053     A
GRA       SCT      SPACE  2,10                                                    GRA        1054     A
GRA      *         UPDATE FIELDS IN ENTRY, LINK INSTRUCTION TO *RCT*              GRA        1055     A
GRA                                                                               GRA        1056     A
GRA       SCT4     SA3    A5+B3       LI = R1 + 3                                 GRA        1057          I
         -FCC0594
GRA                SA1    APF                                                     GRA        1058          I
         -FCC0594
GRA                BX2    X2+X3       CAW = CAW ! [LI]                            GRA        1059          I
         -FCC0594
GRA                SX5    B1                                                      GRA        1060          I
         -FCC0594
GRA                SA4    A3-B1       DI = LI - 1                                 GRA        1061          I
         -FCC0594
GRA       SCT4     SA1    APF                                                     FCC0594      22          I
         -CCGA099
GRA                SA4    A5+2        DI = R1 + 2                                 FCC0594      23          I
         -CCGA099
GRA       SCT4     SA3    =XHO$LCM                                                CCGA099      11          I
         -CCGA100
GRA       SCT4     SA1    APF                                                     CCGA100       9     A
GRA                SA4    A5+2        DI = R1 + 2                                 CCGA099      12     A
GRA                ZR     X3,SCT4.1   IF LCM=D SELECTED                           CCGA099      13          I
         -CCGA100
GRA                LX4    59-D.L2P                                                CCGA099      14          I
         -CCGA100
GRA                PL     X4,SCT4.1   IF NOT L2                                   CCGA099      15          I
         -CCGA100
GRA                LX4    D.L2P-D.XMTP                                            CCGA099      16          I
         -CCGA100
GRA                PL     X1,SCT4.1   IF NOT LEVEL 0                              CCGA099      17          I
         -CCGA100
GRA                SX7    B1                                                      CCGA099      18          I
         -CCGA100
GRA                LX7    CA.EUP                                                  CCGA099      19          I
         -CCGA100
GRA                BX6    X6+X7       EU[RCT(J)] = 1                              CCGA099      20          I
         -CCGA100
GRA                SA6    A6                                                      CCGA099      21          I
         -CCGA100
GRA                EQ     SCT                                                     CCGA099      22          I
         -CCGA100
GRA                                                                               CCGA099      23          I
         -CCGA100
GRA       SCT4.1   SA3    APF                                                     CCGA099      24          I
         -CCGA100
GRA                SA4    A4                                                      CCGA099      25          I
         -CCGA100
GRA                SX5    B1                                                      FCC0594      24     A
GRA                MX3    -D.USESL                                                GRA        1062     A
GRA                BX2    X1+X2       APLN[CAW] = APF ! APLN[CAW]                 GRA        1063     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA          LX4    -D.USESP                                          GRA      1064    A
GRA          LX5    CA.NOCCP    NOCC[CAW] = NOCC[CAW] + 1             GRA      1065    A
GRA          BX1    -X3*X4                                            GRA      1066    A
GRA          LX1    CA.SUSEP                                          GRA      1067    A
GRA          BX5    X1+X5       SUSE[CAW] = SUSE[CAW] + USES[DI]      GRA      1068    A
GRA          IX6    X2+X5                                             GRA      1069    A
GRA          LX4    D.USESP+59-D.INCP                                 GRA      1070    A
GRA          SA6    B2          [EA] = CAW                            GRA      1071    A
GRA          SX7    B2-B5                                             GRA      1072    A
GRA          SA7    A5+B3       [LI] = EA - CO  */ SET INDEX TO *RCT* GRA      1073    A
GRA          PL     X4,SCT      IF ^INC[DI]                           GRA      1074    A
GRA          LX4    D.INCP-D.STP                                      GRA      1075    A
GRA          PL     X4,SCT      IF ^ST[DI]                            GRA      1076    A
GRA                                                                   GRA      1077    A
GRA     *    INCREMENT *ST*, ADD USES OF INC TO CANIDATE ENTRY        GRA      1078    A
GRA                                                                   GRA      1079    A
GRA          LX4    1+D.STP-D.XUP                                     FCC0646    16    A
GRA          SX2    B1                                                FCC0646    17    A
GRA          BX3    X2*X4                                             FCC0646    18    A
GRA          LX3    CA.EUP      EU[CAW] = XU[DI]  */ FORCE *EU* IF *XU* FCC0646  19    A
GRA          BX6    X3+X6                                             FCC0646    20    A
GRA          SA5    A5                                                GRA      1080    A
GRA          SB2    "TB"-B1                                           GRA      1081    A
GRA          BX7    "RN"X5                                            GRA      1082    A
GRA          SA4    B2+X7       DIP = TB-1 + RI[R1]                   GRA      1083    A
GRA          BX3    -X3*X4                                            GRA      1084    A
GRA          SX2    X3-1                                              GRA      1085    A
GRA          LX2    CA.SUSEP                                          GRA      1086    A
GRA          IX6    X2+X6       SUSE[CAW] = SUSE[CAW] + USES[DIP]-1   GRA      1087    A
GRA          SA6    A6                                                GRA      1088    A
GRA                                                                   GRA      1089    A
GRA     *    COLLECT INFORMATION ABOUT THE INCREMENT VALUE OF THE *RD* GRA      1090    A
GRA                                                                   GRA      1091    A
GRA          SA3    A6+2        CCW = EA + 2                          GRA      1092    A
GRA          LX3    59-CC.I1P                                         GRA      1093    A
GRA          SA2    A4+B1       LIP = DIP + 1;  I = CO[LIP]           GRA      1094    A
GRA          PL     X3,SCT5     IF ^I1[CCW]   */ THIS IS THE FIRST INC ST GRA  1095    A
GRA                                                                   GRA      1096    A
GRA          SX4    B1                                                GRA      1097    A
GRA          LX4    CC.I2P                                            GRA      1098    A
GRA          LX3    1+CC.I1P                                          GRA      1099    A
GRA          BX7    X4+X3       I2[CCW] = 1   */ INDICATE MORE THAN 1 INC GRA  1100    A
GRA          SA7    A3                                                GRA      1101    A
GRA          BX3    X7                                                GRA      1102    A
GRA          ZR     X2,SCT6     IF I = 0      */ INCOP IS IA/IS       GRA      1103    A
GRA          EQ     SCT                                               GRA      1104    A
GRA                                                                   GRA      1105    A
GRA     SCT5 LX3    1+CC.I1P                                          GRA      1106    A
GRA          ZR     X2,SCT6     IF I = 0      */ INCOP IS IA OR IS    GRA      1107    A
GRA     SCT  SPACE  1,10                                              GRA      1108    A
GRA     *    INCOP IS AN *STT*, SET INC SIGN                          GRA      1109    A
GRA                                                                   GRA      1110    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SA1    A4-B1       R2P = DIP - 1                                GRA        1111     A
GRA                LX1    42-IH.CAP         SIGN = IF( CA[R2P] < 377777B )         GRA        1112     A
GRA                EQ     SCT10                    THEN 0; ELSE 1                  GRA        1113     A
GRA                                                                               GRA        1114     A
GRA      *         INCOP IS *IA* OR *IS*                                          GRA        1115     A
GRA                                                                               GRA        1116     A
GRA       SCT6     SA1    A4-2        R1P = DIP - 2                                GRA        1117     A
GRA                UX5    B2,X1                                                    GRA        1118     A
GRA                LX5    -R1.RJP                                                  GRA        1119     A
GRA                BX7    "RN"X5      R = RJ[R1]                                   GRA        1120     A
GRA                SA2    "TB"+X7     LIP = TB + R;  I = CO[LIP]                   GRA        1121     A
GRA                SA4    B5+X2       CJP = RCT([LIP])                             GRA        1122     A
GRA                LX4    59-CA.DEFP                                               GRA        1123     A
GRA                PL     X4,SCT7     IF ^DEF[CJP]  */ THIS IS THE INC             GRA        1124     A
GRA                LX5    R1.RJP-R1.RKP                                            GRA        1125     A
GRA                BX7    "RN"X5                                                   GRA        1126     A
GRA                SA2    "TB"+X7     LIP = TB + RK[R1];  I = CO[LIP]              GRA        1127     A
GRA                SX1    B2-OC.IA                                                 GRA        1128     A
GRA                LX1    59          SIGN = IF( OC[R1P] = OC.IA ) THEN 0 ELSE 1   GRA        1129     A
GRA                                                                               GRA        1130     A
GRA      *         ADD ENTRY TO *IA* LIST FOR CANIDATE AND INC VALUE                GRA        1131     A
GRA                                                                               GRA        1132     A
GRA       SCT7     SA4    IAI                                                      GRA        1133     A
GRA                ZR     X2,SCT      IF I = 0      */ INC VALUE NOT A CANIDATE    FCC0414       5          I
          -FCC1039
GRA                SX7    X4-77B                                                   GRA        1134          I
          -FCC1039
GRA                MI     X7,SCT8     IF IAI < 77B  */ NO OVERFLOW                 GRA        1135          I
          -FCC1039
GRA                SX7    A6-B5       J = EA - C0   */ SET CANIDATE ORDINAL        FCC1039      18     A
GRA                SA7    A1+B3       [R1+3] = J    */ IN LINK WORD OF IA/IS       FCC1039      19     A
GRA                ZR     X2,SCT12    IF I = 0      */ INC VALUE NOT A CANIDATE    FCC1039      20     A
GRA                SX5    X4-77B                                                   FCC1039      21     A
GRA                MI     X5,SCT8     IF IAI < 77B  */ NO OVERFLOW                 FCC1039      22     A
GRA                                                                               GRA        1136     A
GRA                SA4    B5+X2                                                    GRA        1137     A
GRA                SX7    B1                                                       GRA        1138     A
GRA                LX7    CA.EUP                                                   GRA        1139     A
GRA                BX7    X7+X4       EU[RCT(I)] = 1                               GRA        1140     A
GRA                SA7    A4                                                       GRA        1141     A
GRA                EQ     SCT9                                                     GRA        1142     A
GRA                                                                               GRA        1143     A
GRA       SCT8     SX7    A6-B5       J = EA - C0   */ CANIDATE ORDINAL            GRA        1144          I
          -FCC1039
GRA                SA7    A1+B3       [R1P+3] = J   */ SET CANIDATE ORD            GRA        1145          I
          -FCC1039
GRA                LX2    18                                                       GRA        1146          I
          -FCC1039
GRA       SCT8     LX2    18                                                       FCC1039      23     A
GRA                BX7    X2+X7                                                    GRA        1147     A
GRA                SA7    SCR+X4      SCR(IAI) = SHIFT(I,18) ! J                   GRA        1148     A
GRA                LX2    -18                                                      GRA        1149     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SX7    X4+B1       IAI = IAI + 1                                  GRA          1150     A
GRA                SA7    A4                                                         GRA          1151     A
GRA                                                                                  GRA          1152     A
GRA        SCT9    BX4    X3                                                         GRA          1153     A
GRA                LX4    59-CC.I2P                                                  GRA          1154     A
GRA                MI     X4,SCT      IF I2[CCW]                                     GRA          1155     A
GRA                                                                                  GRA          1156     A
GRA       *        SAVE *INC* INFORMATION IN *CCW* OF CANIDATE ENTRY                 GRA          1157     A
GRA                                                                                  GRA          1158     A
GRA        SCT10   MX4    1                                                          GRA          1159     A
GRA                BX1    X4*X1       IM[CCW] = SIGN                                 GRA          1160     A
GRA                LX1    1+CC.IMP                                                   GRA          1161     A
GRA                LX4    1+CC.I1P    I1[CCW] = 1                                    GRA          1162     A
GRA                BX4    X1+X4                                                      GRA          1163     A
GRA                SA1    APF                                                        FCC0603        17     A
GRA                LX2    CC.INCP                                                    GRA          1164     A
GRA                IX3    X2+X3       INC[CCW] = I                                   GRA          1165     A
GRA                BX7    X4+X3                                                      GRA          1166     A
GRA                SA7    A3                                                         GRA          1167          I
          -FCC0603
GRA                NZ     X1,SCT11    IF APF " 0    */ INCR UNCONDITINALLY EXEC      FCC0603        18     A
GRA                SX4    B1                                                         FCC0603        19     A
GRA                LX4    CC.I2P                                                     FCC0603        20     A
GRA                BX7    X4+X7       I2[CCW] = 1   */ INHIBIT PREFETCHING           FCC0603        21     A
GRA        SCT11   SA7    A3                                                         FCC0603        22     A
GRA                EQ     SCT                                                        GRA          1168     A
GRA                                                                                  FCC1039        24     A
GRA       *        INC VALUE NOT A CANDIDATE ( LVL2 / LU ) , SET *EU* FOR INCR       FCC1039        25     A
GRA                                                                                  FCC1039        26     A
GRA        SCT12   SX7    B1                                                         FCC1039        27     A
GRA                LX7    CA.EUP                                                     FCC1039        28     A
GRA                BX6    X6+X7       EU[RCT(J)] = 1                                 FCC1039        29     A
GRA                SA6    A6                                                         FCC1039        30     A
GRA                EQ     SCT                                                        FCC1039        31     A
GRA                                                                                  GRA          1169     A
GRA        SCTA    BFMW   CC,(H2,MIT)                                                FCC1014        81     A
GRA                                                                                  FCC1014        82     A
GRA                QUAL   *                                                          GRA          1170     A
GRA        ERC     EQU    /ERC/ERC                                                  GRA          1171     A
GRA        DUM     TITLE  DUM - DETERMINE USEAGE MODE OF INSTRUCTIONS               GRA          1172     A
GRA       **       DUM - DETERMINE USEAGE MODE AND CANIDATE TYPE OF INSTRUCTIONS.    GRA          1173     A
GRA       *                                                                          GRA          1174     A
GRA       *        A BACKWARD SCAN OVER THE BLOCKS TO SET THE USEAGE MODE BITS       GRA          1175     A
GRA       *        ( EU, RF, TU, LU ) AND THE CANIDATE TYPES OF THE MEMORY REFS.     GRA          1176     A
GRA       *                                                                          GRA          1177     A
GRA       *        NOTE THAT THE *IST* BIT IN *UDT* IS ON IF A VARIABLE WAS          GRA          1178     A
GRA       *        REFERENCED IN A INNERMOST LOOP OR THEIR WAS AN INTERFERING        GRA          1179     A
GRA       *        STORE INTO THE CLASS BASE IN THIS LOOP.                           GRA          1180     A
GRA       *                                                                          GRA          1181     A
GRA       *        EXIT   (A0) = NC , MAX NUMBER OF REGISTER CANIDATES               GRA          1182     A
GRA                                                                                  GRA          1183     A
GRA        DUMA    BFMW   CA,(EU,RF)                                                 GRA          1184     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA        DUMB     BFMW    CA,(EU,^RD)                                          GRA        1185     A
GRA        DUMC     BFMW    CA,(EU,RF,TU)                                        GRA        1186          I
          -FCC0735
GRA        DUMC     BFMW    CA,(EU,TU)                                           FCC0735       6     A
GRA                                                                              GRA        1187     A
GRA                 QUAL    DUM                                                  GRA        1188     A
GRA                                                                              GRA        1189     A
GRA                 PROCESS BOS                                                  GRA        1190     A
GRA                 SX6     A0+         BSW = NC                                 GRA        1191     A
GRA                 SA6     =XBSW                                                GRA        1192     A
GRA        DUM      ROUTINE                                                      GRA        1193     A
GRA                 RJ      FBM         FIND BOUNDARY MARKERS                    FCC2124       1     A
GRA                 SA2     O.UDT                                                GRA        1194     A
GRA                 SA4     UVA                                                  GRA        1195     A
GRA                 SB4     59-UD.ISTP                                           GRA        1196     A
GRA                 S"TB"   X5+3        TB = O.SEQ + 3                           GRA        1197     A
GRA                 SB5     X4                                                   GRA        1198     A
GRA                 SB6     X2                                                   GRA        1199     A
GRA                 SA3     X5+B1       R2 = O.SEQ + 1 */ R2 OF *BOS*            GRA        1200     A
GRA                 LX3     -R2.TXTLP                                            GRA        1201     A
GRA                 IX6     X5+X3                                                GRA        1202     A
GRA                 MX7     D.TYL                                                GRA        1203     A
GRA                 SA0     -B1         NC = -1       */ N. CANIDATES            GRA        1204     A
GRA                 LX7     D.TYL+D.TYP                                          GRA        1205     A
GRA                 SA5     X6-4        R1 = O.SEQ + TXTL[R2] - 4                GRA        1206     A
GRA                                                                              GRA        1207     A
GRA       *         SHORT CONSTANTS, BUMP NUMBER OF CANIDATES                    GRA        1208     A
GRA                                                                              GRA        1209     A
GRA                 PROCESS (S,FMA,CLR)                                          GRA        1210     A
GRA                 SA0     A0+1        NC = NC + 1                              GRA        1211     A
GRA                                                                              GRA        1212     A
GRA       *         ADVANCE TO NEXT INSTRUCTION, JUMP TO OPCODE PROCESSOR        GRA        1213     A
GRA                                                                              GRA        1214     A
GRA        DUM1     SA4     A5-2        DI = R1 - 2                              GRA        1215     A
GRA                 SA5     A4-2        R1 = DI - 2                              GRA        1216     A
GRA                 BX3     X7*X4                                                GRA        1217     A
GRA                 UX6     B2,X5                                                GRA        1218     A
GRA                 SA1     DUM.JT+B2                                            GRA        1219     A
GRA                 NZ      X3,DUM2     IF TYPE[DI] " I                          GRA        1220     A
GRA                 LX6     -R1.RJP                                              GRA        1221     A
GRA                 BX3     "RN"X6                                               GRA        1222     A
GRA                 LX6     R1.RJP-R1.RKP                                        GRA        1223     A
GRA                 SA2     "TB"+X3     LJ = TB + RJ[R1]                         GRA        1224     A
GRA                 BX6     "RN"X6                                               GRA        1225     A
GRA                 SA3     "TB"+X6     LK = TB + RK[R1]                         GRA        1226     A
GRA                                                                              GRA        1227     A
GRA        DUM2     SB2     X1                                                   GRA        1228     A
GRA                 JP      B2          JUMP( [MTT.JT( OC[R1] )] )               GRA        1229     A
GRA        TYI      SPACE   3,14                                                 GRA        1230     A
GRA       *         TYI - GENERAL TYPE I INSTRUCTION                             GRA        1231     A
GRA                                                                              GRA        1232     A
GRA        TYI      SX1     B1                                                   GRA        1233     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                LX1    CA.EUP                                                GRA        1234     A
GRA                BX6    X1+X2       EU[LJ] = 1                                GRA        1235     A
GRA                SA6    A2                                                    GRA        1236     A
GRA                BX6    X1+X3       EU[LK] = 1                                GRA        1237     A
GRA                SA6    A3                                                    GRA        1238     A
GRA                EQ     DUM1                                                  GRA        1239     A
GRA        XMT     SPACE  3,8                                                   GRA        1240     A
GRA       *        XMT, XMTC - TRANSMIT USEAGE MODE BACK TO OPERAND              GRA        1241          I
          -CCG0037
GRA       *        XMT - TRANSMIT USAGE MODE BACK TO OPERAND                     CCG0037       5     A
GRA                                                                             GRA        1242     A
GRA                PROCESS (XMT,XMTC)                                            GRA        1243          I
          -CCG0037
GRA                PROCESS  XMT                                                  CCG0037       6     A
GRA                SA1    A4+B1       LI = DI + 1                               GRA        1244     A
GRA                BX6    X1+X2       [LJ] = [LI] ! [LJ]                        GRA        1245     A
GRA                SA6    A2+                                                   GRA        1246     A
GRA                EQ     DUM1                                                  GRA        1247     A
GRA        SA      SPACE  2                                                     GRA        1248     A
GRA       *        SA,SS - INDICATE SHORT USEAGE                                 GRA        1249     A
GRA                                                                             GRA        1250     A
GRA                PROCESS (SA,SS)                                               GRA        1251     A
GRA                SA1    DUMA                                                  GRA        1252          I
          -FCC2124
GRA                BX6    X1+X2       (EU,RF)[LJ] = 1                           GRA        1253          I
          -FCC2124
GRA                SA6    A2                                                    GRA        1254          I
          -FCC2124
GRA                BX6    X1+X3       (EU,RF)[LK] = 1                           GRA        1255          I
          -FCC2124
GRA                SA6    A3                                                    GRA        1256          I
          -FCC2124
GRA                SA1    DUMA        BITS = (EU,RF)                   000330   FCC2124       2     A
GRA                SA4    LBM                                          000340   FCC2124       3     A
GRA                IX6    X4-X4                                        000350   FCC2124       4     A
GRA                PL     X6,SA1      IF RK \ LBM                      000360   FCC2124       5     A
GRA                SX1    B1                                           000370   FCC2124       6     A
GRA                LX1    CA.EUP      BITS = EU                        000380   FCC2124       7     A
GRA        SA1     BX6    X1+X3       BITS[LK] = 1                     000390   FCC2124       8     A
GRA                SA6    A3                                           000400   FCC2124       9     A
GRA                SA1    DUMA        BITS = (EU,RF)                   000410   FCC2124      10     A
GRA                SX5    A2-"TB"                                      000420   FCC2124      11     A
GRA                IX6    X5-X4                                        000430   FCC2124      12     A
GRA                PL     X6,SA2      IF RJ \ LBM                      000440   FCC2124      13     A
GRA                SX1    B1                                           000450   FCC2124      14     A
GRA                LX1    CA.EUP      BITS = EU                        000460   FCC2124      15     A
GRA        SA2     BX6    X1+X2       BITS[LJ] = 1                     000470   FCC2124      16     A
GRA                SA6    A2                                           000480   FCC2124      17     A
GRA                EQ     DUM1                                                  GRA        1257     A
GRA                                                                             GRA        1258     A
GRA       *        TYPE I, B AND X OPERANDS                                      GRA        1259     A
GRA                                                                             GRA        1260     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                 PROCESS (ILS,IRS,PK)                                                GRA         1261     A
GRA                 SA1    DUMA                                                         GRA         1262          I
          -FCC2124
GRA                 BX6    X1+X2       (RF,EU)[LJ] = 1                                  GRA         1263          I
          -FCC2124
GRA                 SA6    A2                                                           GRA         1264          I
          -FCC2124
GRA                 SA1    DUMA        BITS = (EU,RF)                       000500      FCC2124       18     A
GRA                 SA4    LBM                                              000510      FCC2124       19     A
GRA                 SX5    A2-"TB"                                          000520      FCC2124       20     A
GRA                 IX6    X5-X4                                            000530      FCC2124       21     A
GRA                 PL     X6,ILS1     IF RJ \ LBM                          000540      FCC2124       22     A
GRA                 SX1    B1                                               000550      FCC2124       23     A
GRA                 LX1    CA.EUP      BITS = EU                            000560      FCC2124       24     A
GRA       ILS1      BX6    X1+X2                                            000570      FCC2124       25     A
GRA                 SA6    A2          BITS[LJ] = 1                         000580      FCC2124       26     A
GRA                                                                                    GRA         1265     A
GRA       *         TYPE I - 1 OPERAND                                                 GRA         1266     A
GRA                                                                                    GRA         1267     A
GRA                 PROCESS (NR,RNZ,UP)                                                GRA         1268     A
GRA                 SX1    B1                                                          GRA         1269     A
GRA                 LX1    CA.EUP                                                      GRA         1270     A
GRA                 BX6    X1+X3       EU[LK] = 1                                      GRA         1271     A
GRA                 SA6    A3                                                          GRA         1272     A
GRA                 EQ     DUM1                                                        GRA         1273     A
GRA                                                                                    GRA         1274     A
GRA       *         DIRECT READ/WRITE INSTRUCTIONS                                     GRA         1275     A
GRA                                                                                    GRA         1276     A
GRA                 PROCESS DWL                                                        GRA         1277     A
GRA                 BX6    "RN"X5                                                      GRA         1278     A
GRA                 SA3    "TB"+X6     LI = TB + RI[R1]                                GRA         1279     A
GRA                 SA1    DUMB                                                        GRA         1280     A
GRA                 BX6    X1+X3       (EU,^RD)[LI] = 1                                GRA         1281     A
GRA                 SA6    A3                                                          GRA         1282     A
GRA                                                                                    GRA         1283     A
GRA                 PROCESS DRL                                                        GRA         1284          I
          -CCG0037
GRA                 PROCESS  (XMTC,DRL)                                                CCG0037        7     A
GRA                 SX1    B1                                                          GRA         1285     A
GRA                 LX1    CA.EUP                                                      GRA         1286     A
GRA                 BX6    X1+X2       EU[LJ] = 1                                      GRA         1287     A
GRA                 SA6    A2                                                          GRA         1288     A
GRA                 EQ     DUM1                                                        GRA         1289     A
GRA        IA       SPACE  3,14                                                        GRA         1290     A
GRA       *         IA,IS - PROPAGATE *EU,RF,TU* BITS DOWN                             GRA         1291          I
          -FCC0735
GRA       *         SET *IA* BIT, SO WE CAN SET THE *EU* BIT LATER IF NECESSARY        GRA         1292          I
          -FCC0735
GRA                                                                                    GRA         1293          I
          -FCC0735
GRA                 PROCESS (IA,IS)                                                    GRA         1294          I
          -FCC0735

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                 SA1    A4+B1       LI = DI + 1                                    GRA        1295        I
          -FCC0735
GRA                 SX5    B1                                                         GRA        1296        I
          -FCC0735
GRA                 LX4    59-D.INCP                                                  GRA        1297        I
          -FCC0735
GRA                 LX5    CA.EUP      BIT = EU                                       GRA        1298        I
          -FCC0735
GRA                 PL     X4,IA1      IF ^INC[DI]   */ NOT AN INCREMENT              GRA        1299        I
          -FCC0735
GRA                 SX5    0           BIT = 0                                        GRA        1300        I
          -FCC0735
GRA                                                                                   GRA        1301        I
          -FCC0735
GRA        IA1      SA4    DUMC                                                       GRA        1302        I
          -FCC0735
GRA                 BX1    X4*X1       UMB = (EU,RF,TU)[LI]                           GRA        1303        I
          -FCC0735
GRA                 SX6    B1                                                         GRA        1304        I
          -FCC0735
GRA                 BX5    X1+X5                                                      GRA        1305        I
          -FCC0735
GRA                 LX6    CA.IAP                                                     GRA        1306        I
          -FCC0735
GRA                 BX5    X6+X5       IA[UMB] = 1                                    GRA        1307        I
          -FCC0735
GRA                 BX6    X5+X2       BIT[LJ] = 1;  (EU,RF,TU)[LJ] = UMB             GRA        1308        I
          -FCC0735
GRA                 SA6    A2                                                         GRA        1309        I
          -FCC0735
GRA                 BX6    X5+X3       BIT[LK] = 1;  (EU,RF,TU)[LK] = UMB             GRA        1310        I
          -FCC0735
GRA                 SA6    A3                                                         GRA        1311        I
          -FCC0735
GRA       *         IA,IS - PROPAGATE EU,TU AND INC[DI]*RF BITS DOWN, IF ^INC         FCC0735       7     A
GRA       *         THEN SET *EU*, ELSE SET *IA* BIT, SO CAN SET *EU* LATER IF NEC    FCC0735       8     A
GRA                                                                                   FCC0735       9     A
GRA                 PROCESS (IA,IS)                                                   FCC0735      10     A
GRA                 SA1    A4+B1       LI = DI + 1                                    FCC0735      11     A
GRA                 SX5    B1                                                         FCC0735      12     A
GRA                 LX4    -D.INCP                                                    FCC0735      13     A
GRA                 BX6    X5*X4       I = INC[DI]                                    FCC0735      14     A
GRA                 LX5    CA.EUP      BIT = EU                                       FCC0735      15     A
GRA                 ZR     X6,IA1      IF I = 0                                       FCC0735      16     A
GRA                 LX5    CA.IAP-CA.EUP     BIT = IA                                 FCC0735      17     A
GRA                                                                                   FCC0735      18     A
GRA        IA1      SA4    DUMC                                                       FCC0735      19     A
GRA                 LX6    CA.RFP                                                     FCC0735      20     A
GRA                 BX4    X6+X4                                                      FCC0735      21     A
GRA                 BX1    X4*X1       UMB = (EU,TU,I*RF)[LI]                         FCC0735      22     A
GRA                 BX5    X5+X1                                                      FCC0735      23     A
GRA                 BX6    X5+X2       BIT[LJ] = 1;  [LJ] = [LJ] ! UMB                FCC0735      24     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SA6    A2                                                    FCC0735      25    A
GRA                BX6    X5+X3       BIT[LK] = 1;  [LK] = [LK] ! UMB           FCC0735      26    A
GRA                SA6    A3                                                    FCC0735      27    A
GRA                EQ     DUM1                                                  GRA        1312    A
GRA       JPX      SPACE  3,14                                                  GRA        1313    A
GRA      *         X - JUMPS - PROPAGATE *TU* BIT DOWN IT LOOP TEST JUMP         GRA        1314    A
GRA                                                                             GRA        1315    A
GRA                PROCESS JPX                                                  GRA        1316          I
         -CCG2
GRA                PROCESS (JPX,RJXJ)                                           CCG2         13    A
GRA                SA2    LTT                                                   GRA        1317    A
GRA                BX6    "RN"X5                                                GRA        1318    A
GRA                SX1    B1                                                    GRA        1319    A
GRA                SA3    "TB"+X6     LI = TB + RI[R1]                          GRA        1320    A
GRA                LX1    CA.EUP      BIT = EU                                  GRA        1321    A
GRA                ZR     X2,JPX1     IF LTT = 0    */ NO TEST REPLACEMENT      GRA        1322    A
GRA                                                                             GRA        1323    A
GRA                LX4    59-D.INCP                                             GRA        1324    A
GRA                PL     X4,JPX1     IF ^INC[DI]   */ THIS IS NOT THE TEST JP  GRA        1325    A
GRA                LX1    CA.TUP-CA.EUP     BIT = TU                            GRA        1326    A
GRA                                                                             GRA        1327    A
GRA       JPX1     BX6    X1+X3       BIT[LI] = 1                               GRA        1328    A
GRA                SA6    A3                                                    GRA        1329    A
GRA                SA2    LBM                                          000600   FCC2124      27    A
GRA                ZR     X2,DUM1     IF LBM = 0    */ NOT DOING RF CHECKING    000610   FCC2124      28    A
GRA                SA3    A4+B1                                        000620   FCC2124      29    A
GRA                SX6    X3          LBM = [DI+1]  */ SET LOW BND FOR SAFE RF 000630   FCC2124      30    A
GRA                SA6    A2                                           000640   FCC2124      31    A
GRA                EQ     DUM1                                                  GRA        1330    A
GRA       STT      SPACE  3,14                                                  GRA        1331    A
GRA      *         STT - SET RF USEAGE AND PROPAGATE *TU* BIT DOWN               GRA        1332    A
GRA      *         PROPAGATE *EU* BIT TO PRED, IF AN *INC* STT                   GRA        1333    A
GRA                                                                             GRA        1334    A
GRA                PROCESS STT                                                  GRA        1335    A
GRA                SA3    A5+B1       R2 = R1 + 1                               GRA        1336    A
GRA                SA0    A0+B1       NC = NC + 1                               GRA        1337    A
GRA                LX3    -IH.RFP                                               GRA        1338    A
GRA                BX5    "RN"X3      R = RF[R2]                                GRA        1339    A
GRA                ZR     X5,DUM1     IF R = 0      */ NO OPERANDS              GRA        1340    A
GRA                                                                             GRA        1341    A
GRA                SA1    A4+B1       LI = DI + 1                               GRA        1342    A
GRA                LX4    CA.EUP-D.INCP                                         GRA        1343    A
GRA                SA2    "TB"+X5     LJ = TB + R                               GRA        1344    A
GRA                SX6    B1                                                    GRA        1345    A
GRA                LX6    CA.EUP                                                GRA        1346    A
GRA                BX4    X6*X4                                                 GRA        1347    A
GRA                LX6    CA.TUP-CA.EUP                                         GRA        1348    A
GRA                BX3    X6*X1       TU[LJ] = TU[LI]                           GRA        1349          I
         -CCG34
GRA                BX5    X6*X1       TU[LJ] = TU[LI]                      000860   CCG34        10    A
GRA                BX4    X4*X1       EU[LJ] = EU[LI] & INC[DI]                     GRA        1350    A
GRA                LX6    CA.RFP-CA.TUP                                         GRA        1351    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                 BX4    X3+X4                                                GRA        1352          I
          -CCG34
GRA                 BX5    X6+X4       RF[LJ] = 1                               GRA        1353          I
          -CCG34
GRA                 BX6    X5+X2                                                GRA        1354          I
          -CCG34
GRA                 BX4    X4+X5                                       000880   CCG34        11     A
GRA                 BX2    X4+X2                                       000890   CCG34        12     A
GRA                 LX3    IH.RFP-IH.IHP                               000900   CCG34        13     A
GRA       .FTN      IFEQ   HC.ID,2                                     000910   CCG34        14     A
GRA                 SA1    =XS=TRACE                                   000920   CCG34        15     A
GRA                 SX3    X3                                          000930   CCG34        16     A
GRA                 IX1    X1-X3                                       000940   CCG34        17     A
GRA                 ZR     X1,STT1     IF IH[R2] = TRACE.              000950   CCG34        18     A
GRA                 BX2    X6+X2       RF[LJ] = 1                      000960   CCG34        19     A
GRA                                                                    000970   CCG34        20     A
GRA        STT1     BSS    0                                           000980   CCG34        21     A
GRA       .FTN      ELSE                                               000990   CCG34        22     A
GRA                 BX2    X6+X2                                       001000   CCG34        23     A
GRA        .FTN     ENDIF                                              001010   CCG34        24     A
GRA                 BX6    X2                                          001020   CCG34        25     A
GRA                 SA6    A2                                                   GRA        1355     A
GRA                 EQ     DUM1                                                 GRA        1356     A
GRA        RS       SPACE  2,10                                                 GRA        1357     A
GRA       *         DEF, RS - REMOVE SPECIFIED REGISTER FROM AVAILABLE SET       GRA        1358     A
GRA                                                                             GRA        1359     A
GRA                 PROCESS (DEF,RS)                                            GRA        1360          I
          -FCC2056
GRA                 PROCESS RS                                                  FCC2056       7     A
GRA                 BX6    "RN"X5                                               FCC2056       8     A
GRA                 SA2    "TB"+X6     LIP = TB + RI[R1]                        FCC2056       9     A
GRA                 SX1    B1                                                   FCC2056      10     A
GRA                 LX1    CA.EUP                                               FCC2056      11     A
GRA                 BX6    X1+X2       EU[LIP] = 1   */ INHIBIT BIASING         FCC2056      12     A
GRA                 SA6    A2                                                   FCC2056      13     A
GRA                                                                             FCC2056      14     A
GRA                 PROCESS DEF                                                 FCC2056      15     A
GRA                 SA2    MRA                                                  GRA        1361     A
GRA                 LX5    -R1.SOP                                              GRA        1362     A
GRA                 MX3    -SO.REGL                                             GRA        1363     A
GRA                 SX1    B1                                                   GRA        1364     A
GRA                 BX6    -X3*X5                                               GRA        1365     A
GRA                 SB2    X6          REGN = SOREG[R1]                         GRA        1366     A
GRA                 LX1    B2,X1                                                GRA        1367     A
GRA                 BX6    -X1*X2      MRA = MRA & ^SHIFT(1,REGN)               GRA        1368     A
GRA                 LX5    59-SO.LKP                                            GRA        1369     A
GRA                 SA6    A2+                                                  GRA        1370     A
GRA                 PL     X5,DUM1     IF SOLK[R1] " 1 ! 3  */ ^ TEMP/RJRS LOCK GRA        1371     A
GRA                                                                             GRA        1372     A
GRA                 SA2    TRA                                                  GRA        1373     A
GRA                 BX6    X1+X2       TRA = TRA ! SHIFT(1,REGN)                GRA        1374     A
GRA                 SA6    A2                                                   GRA        1375     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                EQ     DUM1                                                    GRA        1376     A
GRA         KLS    SPACE  2                                                       GRA        1377     A
GRA        *       KLS/KRS - SET *RF* USEAGE FOR *SXT* SO *CON* GOES TO A B-REG   GRA        1378     A
GRA                                                                               GRA        1379     A
GRA                PROCESS (KLS,KRS)                                              GRA        1380     A
GRA                SX6    B1                                                      GRA        1381     A
GRA                SA0    A0+B1       NC = NC + 1                                 GRA        1382     A
GRA                LX6    CA.RFP                                                  GRA        1383     A
GRA                SA6    A5-B1       LIP = R1 - 1;  RF[LIP] = 1                  GRA        1384     A
GRA                EQ     DUM1                                                    GRA        1385     A
GRA         LDST   SPACE  2,18                                                    GRA        1386     A
GRA        **      MEMORY REFERENCE PROCESSING - DETERMINE CANIDATE TYPE          GRA        1387     A
GRA                                                                               GRA        1388     A
GRA                PROCESS TST                                                    GRA        1389     A
GRA                SA3    "TB"+X5     LIP = TB + RI[R1]                           GRA        1390          I
           -CCG21
GRA                LX4    59-D.INCP                                               GRA        1391          I
           -CCG21
GRA                SX1    1                                                       GRA        1392          I
           -CCG21
GRA                LX1    CA.RFP                                                  GRA        1393          I
           -CCG21
GRA                MI     X4,TST1     BIT = IF( INC[DI] ) THEN *RF*; ELSE *EU*    GRA        1394          I
           -CCG21
GRA                LX1    CA.EUP-CA.RFP                                           GRA        1395          I
           -CCG21
GRA         TST1   BX6    X1+X3       BIT[LIP] = 1                                GRA        1396          I
           -CCG21
GRA                SA6    A3                                                      GRA        1397          I
           -CCG21
GRA                SA1    A5+B1       R2 = R1 + 1                                 CCG21        53          I
           -CCG28
GRA                SA3    O.TET                                                   CCG21        54          I
           -CCG28
GRA                LX1    -IH.CAP                                                 CCG21        55          I
           -CCG28
GRA                SB2    X3                                                      CCG21        56          I
           -CCG28
GRA                SA2    B2+X1       TW = TET(CA[R2])                            CCG21        57          I
           -CCG28
GRA                SX6    B1                                                      CCG21        58          I
           -CCG28
GRA                LX6    CA.RFP                                                  CCG21        59          I
           -CCG28
GRA                LX2    59-T.RFP                                                CCG21        60          I
           -CCG28
GRA                MI     X2,TST1     BIT = IF( RF[TW] ) THEN *RF*; ELSE *EU*     CCG21        61          I
           -CCG28
GRA                LX6    CA.EUP-CA.RFP                                           CCG21        62          I
           -CCG28
GRA         TST1   SA3    "TB"+X5     LIP = TB + RI[R1]                           CCG21        63          I
           -CCG28

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                BX6    X6+X3                                                    CCG21        64          I
          -CCG28
GRA                SA6    A3                                                       CCG21        65          I
          -CCG28
GRA                LX4    59-D.INCP                                                CCG21        66          I
          -CCG28
GRA                PL     X4,DUM1     IF ^INC[DI]   */ NOT IN A REG IN INNER LP    GRA        1398          I
          -CCG28
GRA                SA1    A5+B1       R2 = R1 + 1                                  CCG28        15     A
GRA                SA2    ITL                                                      CCG28        16     A
GRA                LX1    -IH.CAP                                                  CCG28        17     A
GRA                SB2    X1          C = CA[R2]                                   CCG28        18     A
GRA                SX6    B1                                                       CCG28        19     A
GRA                SB3    X2                                                       CCG28        20     A
GRA                LX6    CA.EUP                                                   CCG28        21     A
GRA                SA2    O.TET                                                    CCG28        22     A
GRA                SA3    "TB"+X5     LIP = TB + RI[R1]                            CCG28        23     A
GRA                SA2    X2+B2       TW = TET(C)                                  CCG28        24     A
GRA                GE     B2,B3,TST1  IF C > ITL    */ CREATED IN THIS LOOP        CCG28        25     A
GRA                BX6    X6+X3       EU[LIP] = 1                                  CCG28        26     A
GRA                SA6    A3                                                       CCG28        27     A
GRA                EQ     DUM1                                                     CCG28        28     A
GRA                                                                               CCG28        29     A
GRA       TST1     LX2    59-T.RFP                                                 CCG28        30          I
          -CCGA009
GRA                PL     X2,TST2     BIT = IF( RF[TW] ) THEN *RF*; ELSE *EU*      CCG28        31          I
          -FCC2124 -CCGA009
GRA                LX6    CA.RFP-CA.EUP                                            CCG28        32          I
          -FCC2124 -CCGA009
GRA       TST2     BX6    X6+X3       BIT[LIP] = 1                                 CCG28        33          I
          -FCC2124 -CCGA009
GRA                SA6    A3                                                       CCG28        34          I
          -FCC2124 -CCGA009
GRA                SB2    A5-B1                                                    FCC1518X     11          I
          -FCC2124 -CCGA009
GRA                PL     X2,TST2     BIT = IF (NOT RF[TW] ) THEN *EU*     001140  FCC2124      32          I
          -CCGA009
GRA       TST1     LX4    59-D.INCP                                                CCGA009       5     A
GRA                MI     X4,TST2     IF (NOT INC[DW])                             CCGA009       6     A
GRA                BX6    X6+X3                                            001150  FCC2124      33     A
GRA                SA6    A3                                               001160  FCC2124      34     A
GRA                EQ     DUM1                                                     CCGA009       7     A
GRA                                                                               CCGA009       8     A
GRA       TST2     SB2    A5-B1                                            001170  FCC2124      35     A
GRA                SB2    A3-B2                                                    FCC1518X     12     A
GRA                ZR     B2,.TLD     IF R1-1 = LIP */ INCR NEXT TO STORE          FCC1518X     13     A
GRA                                                                               FCC1518X     14     A
GRA                SA3    A4+B1       LI = DI + 1                                  FCC1518X     15     A
GRA                LX1    CA.EUP-CA.RFP                                            FCC1518X     16          I
          -CCG28
GRA                SX1    B1                                                       CCG28        35     A
GRA                LX1    CA.EUP                                                   CCG28        36     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                BX6    X1+X3       EU[LI] = 1    */ INHIBIT ADDRESS DIFFERENCING       FCC1518X      17     A
GRA                SA6    A3                                                               FCC1518X      18     A
GRA                                                                                        GRA         1399     A
GRA                PROCESS TLD                                                              GRA         1400     A
GRA                SA3    A4+B1       LI = DI + 1                                           GRA         1401     A
GRA                SX1    B1                                                                GRA         1402     A
GRA                SA0    A0+B1       NC = NC + 1                                           GRA         1403     A
GRA                LX1    CA.VCP                                                            GRA         1404     A
GRA                BX6    X1+X3       VC[LI] = 1                                            GRA         1405     A
GRA                SA6    A3                                                                GRA         1406     A
GRA                EQ     DUM1                                                              GRA         1407     A
GRA        ST      SPACE  2                                                                 GRA         1408     A
GRA                PROCESS ST                                                               GRA         1409     A
GRA                BX2    X4                                                                GRA         1410     A
GRA                SX1    B1                                                                GRA         1411     A
GRA                LX2    59-D.INCP                                                         GRA         1412     A
GRA                SA3    "TB"+X5     LIP = TB + RI[R1]                                     GRA         1413     A
GRA                LX1    CA.EUP                                                            GRA         1414     A
GRA                MI     X2,.LD      IF INC[DI]                                            GRA         1415          I
           -FCC1997
GRA                                                                                        GRA         1416          I
           -FCC1997
GRA                BX6    X1+X3       EU[LIP] = 1                                           GRA         1417          I
           -FCC1997
GRA                PL     X2,ST1      IF ^INC[DI]                                           FCC1997        5     A
GRA                                                                                        FCC1997        6     A
GRA                SB2    A5-B1                                                             FCC1997        7     A
GRA                SB2    A3-B2                                                             FCC1997        8     A
GRA                ZR     B2,.LD      IF LIP = R1-1 */ INC NEXT TO STORE                    FCC1997        9     A
GRA                                                                                        FCC1997       10     A
GRA                BX6    X1+X3       EU[LIP] = 1                                           FCC1997       11     A
GRA                SA6    A3                                                                FCC1997       12     A
GRA                EQ     .LD                                                               FCC1997       13     A
GRA                                                                                        FCC1997       14     A
GRA        ST1     BX6    X1+X3       EU[LIP] = 1                                           FCC1997       15     A
GRA                SA6    A3                                                                GRA         1418     A
GRA                LX1    CA.^RDP-CA.EUP                                                    GRA         1419     A
GRA                SA2    A4+B1       LI = DI + 1                                           GRA         1420     A
GRA                BX6    X1+X2       ^RD[LI] = 1   */ INDICATE NON RECURSIVE DEF           GRA         1421     A
GRA                SA6    A2                                                                GRA         1422     A
GRA                                                                                        GRA         1423     A
GRA                PROCESS (LD,ILD)                                                         GRA         1424          I
           -CCG1
GRA                PROCESS (LD,ILD,LDC)                                                     CCG1          93          I
           -CCG16
GRA                PROCESS (LD,ILD,LDC,LDV)                                                 CCG16        214     A
GRA                SA3    A5+B1       R2 = R1 + 1                                           GRA         1425     A
GRA                LX5    -R1.INP                                                           GRA         1426     A
GRA                SX1    B1                                                                GRA         1427     A
GRA                LX3    -IH.RFP                                                           GRA         1428     A
GRA                LX1    CA.VCP      CT = VC       */ SET FOR VALUE CANIDATE               GRA         1429     A
GRA                BX6    "RN"X3                                                            GRA         1430     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                ZR     X6,LD1      IF RF[R2] = 0                                    GRA        1431     A
GRA                                                                                    GRA        1432     A
GRA                SA2    LBM                                                 000660    FCC2124      36     A
GRA                IX3    X6-X2                                               000670    FCC2124      37     A
GRA                MI     X3,LD0      IF RF < LBM   */ CONDITIONAL REF        000680    FCC2124      38     A
GRA                SA2    "TB"+X6     LJ = TB + RF[R2]                                  GRA        1433     A
GRA                LX1    CA.RFP-CA.VCP                                                 GRA        1434     A
GRA                BX6    X1+X2       RF[LJ] = 1                                        GRA        1435     A
GRA                SA6    A2                                                            GRA        1436     A
GRA                LX1    CA.ACP-CA.RFP     CT = AC                                     GRA        1437          I
          -FCC2124
GRA        LD0     SX1    B1                                                  000700    FCC2124      39     A
GRA                LX1    CA.ACP      CT = AC                                 000710    FCC2124      40     A
GRA                EQ     LD3                                                           GRA        1438     A
GRA                                                                                    GRA        1439     A
GRA       *        CHECK SCALAR LD/ST FOR INTERFERENCE, EXPLICIT USE IN INNER LP       GRA        1440     A
GRA                                                                                    GRA        1441     A
GRA        LD1     SA2    B6+X5       UI = UDT( IN[R1] )                                GRA        1442     A
GRA                LX6    B4,X2                                                         GRA        1443     A
GRA                MI     X6,LD2      IF IST[UI]    */ INTERFERENCE                     GRA        1444     A
GRA                LX6    UD.ISTP-UD.CMP                                                GRA        1445     A
GRA                PL     X6,LD3      IF ^CM[UI]    */ NOT MEMBER OF A CLASS            GRA        1446     A
GRA                                                                                    GRA        1447     A
GRA       *        CHECK TO SEE IF BASE MEMBER WAS USED ( INDEXED *LD* )               GRA        1448     A
GRA                                                                                    GRA        1449     A
GRA                LX6    1+UD.CMP-UD.BMIP                                              GRA        1450     A
GRA                SB2    X6+B1                                                         GRA        1451          I
          -FCC1040
GRA                SA3    B6+B2       U2 = UDT( BMI[UI]+1 )                             GRA        1452          I
          -FCC1040
GRA                SA2    B6+X6       UB = UDT( BMI[UI] )                               FCC1040       5     A
GRA                LX2    59-UD.DEFRP                                                   FCC1040       6     A
GRA                MI     X2,LD2      IF DEFR[UB]   */ BASE DEFINED IN INNER LP         FCC1040       7     A
GRA                SA3    A2+B1       U2 = UB + 1                                       FCC1040       8     A
GRA                UX5    B2,X3                                                         GRA        1453     A
GRA                SB2    B2-59                                                         GRA        1454     A
GRA                SA2    B5+X3       UVW = UVA( WI[UI] )                               GRA        1455          I
          -FCC1040
GRA                SA2    B5+X3       UVW = UVA( WI[U2] )                               FCC1040       9     A
GRA                LX5    -B2,X2                                                        GRA        1456     A
GRA                PL     X5,LD3      IF SHIFT( UVW , 59-BITN[UI] ) > 0                 GRA        1457          I
          -FCC1040
GRA                PL     X5,LD3      IF SHIFT( UVW , 59-BITN[U2] ) > 0                 FCC1040      10     A
GRA                                                                                    GRA        1458     A
GRA                LX6    UD.BMIP+59-UD.DEFRP                                           GRA        1459     A
GRA                PL     X6,LD3      IF ^DEFR[UI]  */ NO DEFS IN LOOP                  GRA        1460     A
GRA                                                                                    GRA        1461     A
GRA       *        ^ VC , CLEAR *INC* BIT OF ST PRED SO AS TO FORCE *EU*               GRA        1462     A
GRA       *        FOR CON PRED OF *IA* INSTRUCTION.                                   GRA        1463     A
GRA                                                                                    GRA        1464     A
GRA        LD2     LX4    59-D.INCP                                                     GRA        1465     A
GRA                PL     X4,LD2A     IF ^INC[DI]   */ ^ INC STORE                      GRA        1466     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SX3    B1                                                    GRA        1467     A
GRA                SA2    A5-2        DIP = R1 - 2                              GRA        1468     A
GRA                LX3    D.INCP                                                GRA        1469     A
GRA                BX6    -X3*X2      INC[DIP] = 0                              GRA        1470     A
GRA                SA6    A2                                                    GRA        1471     A
GRA                                                                             GRA        1472     A
GRA       *        CHECK FOR POSSIBLE ADDRESS CANIDATE                           GRA        1473     A
GRA                                                                             GRA        1474     A
GRA        LD2A    LX4    D.INCP-D.FPP                                          GRA        1475     A
GRA                PL     X4,DUM1     IF ^FP[DI]                                GRA        1476     A
GRA                LX1    CA.ACP-CA.VCP     CT = AC                             GRA        1477     A
GRA                                                                             GRA        1478     A
GRA        LD3     SA3    A4+B1       LI = DI + 1                               GRA        1479     A
GRA                SA0    A0+B1       NC = NC + 1                               GRA        1480     A
GRA                BX6    X1+X3       CT[LI] = 1                                GRA        1481     A
GRA                SA6    A3                                                    GRA        1482     A
GRA                EQ     DUM1                                                  GRA        1483     A
GRA        PST     SPACE  3                                                     CCGA082       6     A
GRA                PROCESS PST                                                  CCGA082       7     A
GRA                BX3    "RN"X5                                                CCGA082       8     A
GRA                SX3    X3-1                                                  CCGA082       9     A
GRA                SA2    "TB"+X3     DI OF PRED                                CCGA082      10     A
GRA                SX1    B1                                                    CCGA082      11     A
GRA                LX1    D.INCP                                                CCGA082      12     A
GRA                BX6    -X1*X2      INC[DIP] = 0                              CCGA082      13     A
GRA                SA6    A2                                                    CCGA082      14     A
GRA                EQ     DUM1                                                  CCGA082      15     A
GRA        FBM     SPACE  4,18                                     000730     FCC2124      41     A
GRA       **       FBM - FIND BOUNDARY MARKERS                     000740     FCC2124      42     A
GRA       *        FOWARD SCAN TO FIND BOUNDARY MARKERS IN AN EBB. 000750     FCC2124      43     A
GRA       *        USED BY PROCESSORS SO AS TO NOT SET THE *RF* BIT WHEN THE 000760     FCC2124      44     A
GRA       *        INSTRUCTION REFERENCING THE RF IS CONDITIONALLY EXECUTED. 000770     FCC2124      45     A
GRA       *        AN ASSIGNMENT OF THE RF TO A B-REGISTER MAY CAUSE AN OVERFLOW 000780     FCC2124      46     A
GRA       *        AT EXECUTION TIME.                              000790     FCC2124      47     A
GRA                                                                000800     FCC2124      48     A
GRA        LBM     BSS    1           LAST BOUNDARY MARKER                     000810     FCC2124      49     A
GRA                                                                000820     FCC2124      50     A
GRA        FBM0    SA7    LBM         LBM = LB                                 000830     FCC2124      51     A
GRA        FBM     ROUTINE                                                     000840     FCC2124      52     A
GRA                SX7    B0          LB = 0                                   000850     FCC2124      53     A
GRA                SA1    =XHO$UO                                             000860     FCC2124      54     A
GRA                NZ     X1,FBM0     IF UO " 0                                000870     FCC2124      55     A
GRA                SA1    =XOPT2                                              000880     FCC2124      56          I
          -CCGA009
GRA                ZR     X1,FBM1     IF OPT " 2                               000890     FCC2124      57          I
          -CCGA009
GRA                SA1    HO$OPT                                                       CCGA009       9     A
GRA                PL     X1,FBM1     IF OPT NE 2                                  CCGA009      10     A
GRA                SA2    X5          R1 = [TXT]                               000900     FCC2124      58     A
GRA                MX3    -2                                                  000910     FCC2124      59     A
GRA                LX2    -R1.H2P                                             000920     FCC2124      60     A
GRA                BX4    -X3*X2                                              000930     FCC2124      61     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                NZ     X4,FBM1     IF AP ! LN    */ BLOCK EXECUTED UNCONDITION000940   FCC2124      62     A
GRA                SX7    377777B     LB = INFINITY                               000950   FCC2124      63     A
GRA                EQ     FBM0                                                    000960   FCC2124      64     A
GRA                                                                               000970   FCC2124      65     A
GRA        FBM1    SB6    X5+2        D0 = TXT + 2                                000980   FCC2124      66     A
GRA                SB2    B1+B1                                                   000990   FCC2124      67     A
GRA                SA4    B6          DI = D0                                     001000   FCC2124      68     A
GRA                                                                               001010   FCC2124      69     A
GRA        FBM2    SA3    A4+B2       R1 = DI + 2                                 001020   FCC2124      70     A
GRA                SA4    A3+B2       DI = R1 + 2                                 001030   FCC2124      71     A
GRA                UX6    B3,X3                                                   001040   FCC2124      72     A
GRA                LX4    59-D.JPP                                                001050   FCC2124      73     A
GRA                ZR     B3,FBM0     IF OC[R1] = 0 */ END OF SEQUENCE            001060   FCC2124      74     A
GRA                PL     X4,FBM2     IF ^JP[DI]                                  001070   FCC2124      75     A
GRA                                                                               001080   FCC2124      76     A
GRA                SA7    A4+B1       [DI+1] = LB   */ SAVE INDEX OF LAST BM      001090   FCC2124      77     A
GRA                SX7    A4-B6       LB = DI - D0                                001100   FCC2124      78     A
GRA                EQ     FBM2                                                    001110   FCC2124      79     A
GRA                                                                                        GRA        1484     A
GRA                QUAL   *                                                                GRA        1485     A
GRA        DUM     EQU    /DUM/DUM                                                         GRA        1486     A
GRA        DUM.JT  EJECT                                                                   GRA        1487     A
GRA       **       DEFINE *OPR* MACRO TO CREATE COMBINED JUMP TABLE FOR *ERC/DUM*          GRA        1488     A
GRA       *        AND A LIST OF OPCODES WHICH MAY PRECEED A *RS* TO A B-REG.              GRA        1489     A
GRA                                                                                        GRA        1490     A
GRA                ECHO   2,X=(S,SA,SS,STT)                                                GRA        1491     A
GRA                NOREF  BRS.X                                                            GRA        1492     A
GRA        BRS.X   EQU    1                                                                GRA        1493     A
GRA                                                                                        GRA        1494     A
GRA                MACRO  OPR,NAM,SS                                                       GRA        1495     A
GRA                NOREF  /DUM/.NAM,/ERC/.NAM                                              GRA        1496     A
GRA       *                                                                                GRA        1497     A
GRA                IF     DEF,/ERC/.NAM,2                                                  GRA        1498     A
GRA                EQ     /ERC/.NAM             NAM                                        GRA        1499     A
GRA                ELSE   1                                                                GRA        1500     A
GRA                EQ     /ERC/ERC1   NAM                                                  GRA        1501     A
GRA       *                                                                                GRA        1502     A
GRA                NOREF  BRS.NAM                                                          GRA        1503     A
GRA                IF     DEF,BRS.NAM,2                                                    GRA        1504     A
GRA       -        VFD    12/1S11                                                          GRA        1505     A
GRA                ELSE   1                                                                GRA        1506     A
GRA       -        VFD    12/0                                                             GRA        1507     A
GRA       *                                                                                GRA        1508     A
GRA        O       IF     DEF,/DUM/.NAM                                                    GRA        1509     A
GRA                VFD 18//DUM/.NAM             NAM                                        GRA        1510     A
GRA        O       ELSE                                                                    GRA        1511     A
GRA        TYP     MICRO  4,1,$SS$                                                         GRA        1512     A
GRA                IFC    EQ,/"TYP"/ /,2                                                   GRA        1513     A
GRA                VFD 18//DUM/TYI    NAM                                                  GRA        1514     A
GRA                SKIP   1                                                                GRA        1515     A
GRA                VFD 18//DUM/DUM1                                                        GRA        1516     A
GRA        O       ENDIF                                                                   GRA        1517     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA      *                                                          GRA      1518     A
GRA               ENDM                                              GRA      1519     A
GRA                                                                 GRA      1520     A
GRA       DUM.JT  BSS    0                                          GRA      1521     A
GRA       ERC.JT  BSS    0                                          GRA      1522     A
GRA      *CALL    OPRDEFS                                           GRA      1523     A
GRA       GRA     TTL    GRA - GLOBAL REGISTER ASSIGNMENT / CANIDATE SELECTION  GRA  1524  A
GRA       MTA     TITLE  MTA - MAKE TENATIVE B-ASSIGNMENTS          GRA      1525     A
GRA      **       MTA - MAKE TENATIVE B-ASSIGNMENTS, MAKE TENATIVE COUNTING  GRA  1526  A
GRA      *        METHOD DECISION ( *TRD* ).                        GRA      1527     A
GRA                                                                 GRA      1528     A
GRA       MTAA    BFMW   CA,(VC,RF)                                 GRA      1529     A
GRA       MTAB    BFMW   CA,(IRA,EU,LX)    NO BIAS BITS ( BY AN *CAIH* )  GRA  1530  A
GRA                                                                 GRA      1531     A
GRA       MTA     ROUTINE                                           GRA      1532     A
GRA               SA2    MRA                                        GRA      1533     A
GRA               SX3    376B                                       GRA      1534     A
GRA               BX4    X3*X2       BR = MRA & 376B  */ B-REGS     GRA      1535     A
GRA               SX5    X4+B1                                      GRA      1536     A
GRA               PX6    X5          ABR = PACK( 0 , BR+1 )         GRA      1537     A
GRA               CX7    X4          NAB = COUNT( BR )              GRA      1538     A
GRA               SA6    ABR                                        GRA      1539     A
GRA               SA7    NAB                                        GRA      1540     A
GRA               SA7    NTA         NTA = NAB                      GRA      1541     A
GRA               ZR     X7,MTA      IF NAB = 0    */ NO B-S AVAIL  GRA      1542     A
GRA                                                                 GRA      1543     A
GRA      *        SELECT VALUE CANIDATES THAT ARE *RF* FROM *RCT* AND FORM A  GRA  1544  A
GRA      *        SORT TABLE.  FORMAT - 42/SORT[CAW],18/RCT INDEX  ( *CO* )  GRA  1545  A
GRA                                                                 GRA      1546     A
GRA               SA1    NVC                                        GRA      1547     A
GRA               ALLOC  RND,X1      ALLOC( RND , NVC )  */ SORT TABLE  GRA  1548  A
GRA               SA4    O.RCT                                      GRA      1549     A
GRA               SA5    L.RCT                                      GRA      1550     A
GRA               MX7    0                                          GRA      1551     A
GRA               SA7    X2          SI = [O.RND];  [SI] = 0        GRA      1552     A
GRA               MX0    -CA.SORTL                                  GRA      1553     A
GRA               SB7    X2+B1       S0 = SI + 1                    GRA      1554     A
GRA               LX0    CA.SORTP                                   GRA      1555     A
GRA               SB6    X4          CO = [O.RCT]                   GRA      1556     A
GRA               SB3    3                                          GRA      1557     A
GRA               SA5    B6+X5       CI = CO + L.RCT                GRA      1558     A
GRA               SA1    MTAA                                       GRA      1559     A
GRA                                                                 GRA      1560     A
GRA       MTA1    SA5    A5-B3       CI = CI - 3                    GRA      1561     A
GRA               ZR     X5,MTA2     IF [CI] = 0   */ END OF TABLE  GRA      1562     A
GRA               BX6    -X1+X5                                     GRA      1563     A
GRA               NZ     X6,MTA1     IF ^( VC[CI] & RF[CI] )        GRA      1564     A
GRA                                                                 GRA      1565     A
GRA               BX3    -X0*X5                                     GRA      1566     A
GRA               LX5    59-CA.^RDP                                 GRA      1567     A
GRA               MI     X5,MTA1     IF ^RD[CI]    */ NON RECURSIVE DEF  GRA  1568  A
GRA                                                                 GRA      1569     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SX4    A5-B6       I = CI - CO                                  GRA        1570    A
GRA                IX7    X3+X4                                                    GRA        1571    A
GRA                SA7    A7+B1       SI = SI + 1;  [SI] = SORT[CI] ! I            GRA        1572    A
GRA                EQ     MTA1                                                     GRA        1573    A
GRA                                                                               GRA        1574    A
GRA       MTA2     SB3    A7+B1       SI = SI + 1                                  GRA        1575    A
GRA                SX1    B3-B7       LEN = SI - SO                                GRA        1576    A
GRA                ZR     X1,MTA12    IF LEN = 0    */ NO CANIDATES                GRA        1577    A
GRA                                                                               GRA        1578    A
GRA                CALL   SHL         SORT( RND )                                  GRA        1579          I
          -CCG1
GRA                CALL   SHL=        SORT( RND )                                  CCG1         94          I
          -CCGNC
GRA                CALL   SHL#        SORT( RND )                                  CCGNC       649    A
GRA                TRACE  MTA,RND                                                  GRA        1580    A
GRA                                                                               GRA        1581    A
GRA       *        ON EXIT FROM *SHL* (B6) = CO IS INTACT AND (B3) = SI, LWA+1     GRA        1582    A
GRA       MTA      SPACE  2,14                                                     GRA        1583    A
GRA       *        NOTE THAT THE LOOP INDEX AND LIMIT ( IF A VARIABLE ) WILL BE    GRA        1584    A
GRA       *        THE LAST 2 ENTRIES IN THE TABLE.  USING THESE FACTS AND OTHER   GRA        1585    A
GRA       *        INFO MAKE A PRELIMINARY COUNTING METHOD/TEST REPLACEMENT        GRA        1586    A
GRA       *        DECISION.  SET *TRD* TO ONE OF THE FOLLOWING.                   GRA        1587    A
GRA       *        0      NO TEST REPLACEMENT                                      GRA        1588    A
GRA       *        1      I-N IN A B-REG                                           GRA        1589    A
GRA       *        2      I AND N IN B-REGS                                        GRA        1590    A
GRA       *        3      I, N ALLOCATED, COUNT BY A+I AND LIMIT = A+N             GRA        1591    A
GRA       *        4      I ASSIGNED, COUNT BY F(I), F(N) WHERE *F* IS A           GRA        1592    A
GRA       *                LINEAR FUNCTION OF I ( I.E. A INTEGER POLYNOMIAL ).    GRA        1593    A
GRA                                                                               GRA        1594    A
GRA                SA1    LTT                                                      GRA        1595    A
GRA                SB7    B3-B1       SI = SI - 1   */ LWA OF SORT TABLE           GRA        1596    A
GRA                SA0    B0          TRM = 0       */ TEST REPLACEMENT MODE       GRA        1597    A
GRA                ZR     X1,MTA11    IF LTT = 0    */ LOOP TEST NOT STANDARD FOR  GRA        1598    A
GRA                                                                               GRA        1599    A
GRA       *        GET A B-REG FOR THE *CV*                                        GRA        1600    A
GRA                                                                               GRA        1601    A
GRA                SA5    B7                                                       GRA        1602    A
GRA                SB4    X5          I = CO[SI]    */ CANIDATE ORDINAL            GRA        1603    A
GRA                SA4    B6+X5       CI = RCT(I)                                  GRA        1604    A
GRA                LX4    59-CA.TUP                                                GRA        1605    A
GRA                PL     X4,MTA11    IF ^TU[CI]    */ CHECK FOR *CV*              GRA        1606    A
GRA                LX4    CA.TUP-CA.DEFP                                           GRA        1607    A
GRA                PL     X4,MTA11    IF ^DEF[CI]                                  GRA        1608    A
GRA                MX1    0           INVF = 0                                     GRA        1609    A
GRA                RJ     AIR         ASSIGN *CV* TO AN INDEX REGISTER             GRA        1610    A
GRA                SX6    -B2                                                      GRA        1611    A
GRA                SA6    LCV+1       LCV(2) = REGNO                               GRA        1612    A
GRA                SB7    B7-B1       SI = SI - 1   */ REMOVE FROM SORT TABLE      GRA        1613    A
GRA                AX6    3                                                        GRA        1614    A
GRA                NZ     X6,MTA11    IF REGNO > 10B  */ *CV* IN AN X-REG          GRA        1615    A
GRA       *                          IN FUTURE - DELETE X-ASSIGNMENT AND LOOP     GRA        1616    A
GRA                                                                               GRA        1617    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA      *         FIND THE UPPER LIMIT. IT IS NOT IN *RCT* IF IN A INNER LP          GRA    1618    A
GRA      *         AND IT IS A CONSTANT.                                              GRA    1619    A
GRA                                                                                   GRA    1620    A
GRA       MTA2A    SA5    B7                                                          GRA    1621    A
GRA                SA1    LUL                                                         GRA    1622    A
GRA                SB4    X5          I = CO[SI]    */ POSSIBLE *UL*                  GRA    1623    A
GRA                LX5    59-CA.TUP                                                   GRA    1624    A
GRA                PL     X5,MTA3     IF ^TU[SI]    */ NOT TEST USE                   GRA    1625    A
GRA                                                                                   GRA    1626    A
GRA      *         ENTRY IS *TU*, ENSURE THAT IT IS NOT A VARIABLE INCREMENT          GRA    1627    A
GRA                                                                                   GRA    1628    A
GRA                SA4    B6+B4       CAW = RCT(I)                                    GRA    1629    A
GRA                UX6    B2,X1                                                       GRA    1630    A
GRA                AX6    PS.UDIP                                                     GRA    1631    A
GRA                BX7    X6-X4                                                       GRA    1632    A
GRA                SB3    X7                                                          GRA    1633    A
GRA                ZR     B3,MTA5     IF UDI[CAW] = UDI[LUL]                          GRA    1634    A
GRA                                                                                   GRA    1635    A
GRA                MX2    1                                                           GRA    1636    A
GRA                BX6    -X2*X5      TU[SI] = 0                                      GRA    1637    A
GRA                MX7    0                                                           GRA    1638    A
GRA                LX6    1+CA.TUP                                                    GRA    1639    A
GRA                SA7    STP         STP = 0       */ INHIBIT BIASING IN *MFA*       GRA    1640    A
GRA                SA6    A5                                                          GRA    1641    A
GRA                LX2    1+CA.EUP                                                    GRA    1642    A
GRA                BX7    X2+X4       EU[CAW] = 1   */ INHIBTI BIASING IN *MFA*       GRA    1643    A
GRA                SA7    A4                                                          GRA    1644    A
GRA                ZR     B2,MTA3     IF TYP[LUL] = 0  */ *UL* IS A CON               GRA    1645    A
GRA                                                                                   GRA    1646    A
GRA      *         VARIABLE INC BEFORE VARIABLE *UL*, SWAP ENTRIES                    GRA    1647    A
GRA                                                                                   GRA    1648    A
GRA                SA3    B7-B1                                                       GRA    1649    A
GRA                ZR     X3,MTA11    IF [SI-1] = 0 */ UL NOT IN SORT TABLE           GRA    1650    A
GRA                SA6    A3          SWAP( [SI] , [SI-1] )                           GRA    1651    A
GRA                BX7    X3                                                          GRA    1652    A
GRA                SA7    A5                                                          GRA    1653    A
GRA                EQ     MTA2A                                                       GRA    1654    A
GRA                                                                                   GRA    1655    A
GRA      *         SEARCH *RCT* FOR CON *UL* IF FINAL B-ASSIGNMENTS WERE MADE         GRA    1656    A
GRA                                                                                   GRA    1657    A
GRA       MTA3     UX6    B2,X1                                                       GRA    1658    A
GRA                NZ     B2,MTA11    IF TYP[LUL] " 0   */ *UL* IS A VAR              GRA    1659    A
GRA                LX6    IH.CAP                                                      GRA    1660    A
GRA                SA3    FBA                                                         GRA    1661    A
GRA                SB4    0           I = 0                                           GRA    1662    A
GRA                ZR     X3,MTA5     IF FBA = 0    */ NO FINAL B-ASSIGNMENTS         GRA    1663    A
GRA                                                                                   GRA    1664    A
GRA                SA2    L.RCT                                                       GRA    1665    A
GRA                SB2    B6+B1                                                       GRA    1666    A
GRA                SA6    B6+B1       KEY = [CO+1] = IHW(0,0,CA[LUL],0)               GRA    1667    A
GRA                SA5    B2+X2       CI = CO+1 + L.RCT                               GRA    1668    A
GRA                                                                                   GRA    1669    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA       MTA4     SA5    A5-3        CI = CI - 3                                    GRA     1670    A
GRA                BX7    X6-X5                                                      GRA     1671    A
GRA                NZ     X7,MTA4     IF [CA] " KEY                                  GRA     1672    A
GRA                                                                                  GRA     1673    A
GRA                SB4    A5-B2       I = CI - (CO+1)                                GRA     1674    A
GRA                SA7    B6+B1       [CO+1] = 0                                     GRA     1675    A
GRA       MTA      SPACE  2,10                                                       GRA     1676    A
GRA      *         NOW LOOK AT TEST REPLACEMENT POSSABILITIES                        GRA     1677    A
GRA                                                                                  GRA     1678    A
GRA       MTA5     SA2    B6+B4       ULW = RCT(I)  */ *UL* WORD                     GRA     1679    A
GRA                SA0    B1+B1       TRM = 2       */ I,N IN B-REGS                 GRA     1680    A
GRA                LX2    59-CA.RAP                                                  GRA     1681    A
GRA                PL     X2,MTA6     IF ^RA[ULW]                                    GRA     1682    A
GRA                                                                                  GRA     1683    A
GRA      *         *UL* IN A REGISTER, SET *EU* BIT FOR *CV* SO IT ISNT BIASED.      GRA     1684    A
GRA                                                                                  GRA     1685    A
GRA                SA1    A2+2        CC = RCT(I+2)                                  GRA     1686    A
GRA                LX1    -CC.REG1P                                                  GRA     1687    A
GRA                MX7    -CC.REG1L                                                  GRA     1688    A
GRA                BX6    -X7*X1                                                     GRA     1689    A
GRA                SB2    X6-10B                                                     GRA     1690    A
GRA                SA6    LUL+1       LUL(2) = REG1[CC]                              GRA     1691    A
GRA                SA2    B7+B1       CVO = CO[SI+1]                                 GRA     1692    A
GRA                SA3    B6+X2                                                      GRA     1693    A
GRA                SX1    B1                                                         GRA     1694    A
GRA                LX1    CA.EUP                                                     GRA     1695    A
GRA                BX6    X1+X3       EU[RCT(CVO)] = 1  */ INHIBIT BIASING IN *MFA*  GRA     1696    A
GRA                SA6    A3                                                         GRA     1697    A
GRA                MI     B2,MTA11    IF REG1[CC] < 10B  */ *UL* IN A B-REG          GRA     1698    A
GRA                SA0    B0          TRM = 0                                        GRA     1699    A
GRA                EQ     MTA11                                                      GRA     1700    A
GRA                                                                                  GRA     1701    A
GRA       MTA6     SA1    B7+B1                                                      GRA     1702    A
GRA                SA3    MTAB                                                       GRA     1703    A
GRA                LX2    CA.RAP-CA.SUSEP                                            GRA     1704    A
GRA                SA5    B6+X1       CVW = RCT([SI+1])  */ *CV* WORD                GRA     1705    A
GRA                BX6    X3*X5                                                      GRA     1706    A
GRA                NZ     X6,MTA10    IF (IRA,EU,LX)[CVW]                            GRA     1707    A
GRA                MX7    -CA.SUSEL+1                                                GRA     1708    A
GRA                BX6    -X7*X2                                                     GRA     1709    A
GRA                NZ     X6,MFA6A    IF SUSE[ULW] > 1                               GRA     1710    A
GRA                BX6    X2                                                         GRA     1711    A
GRA                LX6    CA.SUSEP-CA.CONP                                           GRA     1712    A
GRA                PL     X6,MTA7     IF ^CON[ULW]  */ NO OTHER USES                 GRA     1713    A
GRA                                                                                  GRA     1714    A
GRA       MFA6A    SX6    B1                                                         GRA     1715    A
GRA                LX2    1+CA.SUSEP                                                 GRA     1716    A
GRA                LX6    CA.LXP                                                     GRA     1717    A
GRA                BX7    X6+X2       LX[ULW] = 1   */ INHIBIT BIASING IN *MFA*      GRA     1718    A
GRA                SA7    A2                                                         GRA     1719    A
GRA                EQ     MTA10                                                      GRA     1720    A
GRA                                                                                  GRA     1721    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA      *         *CV* IS DEAD ON EXIT AND ^EU                                  GRA      1722     A
GRA                                                                              GRA      1723     A
GRA       MTA7     SA1    TUD                                                    GRA      1724     A
GRA                SA0    A0+B1       TRM = 3       */ A+I, A+N                  GRA      1725     A
GRA                AX7    B1,X1                                                  GRA      1726     A
GRA                NZ     X7,MTA10    IF TUD > 1    */ I IS USED IN INDEXED LD"S GRA      1727     A
GRA                                                                              GRA      1728     A
GRA                SA3    STP                                                    GRA      1729     A
GRA                SA0    1           TRM = 1       */ I-N CASE                  GRA      1730     A
GRA                ZR     X3,MTA8     IF STP = 0    */ NO *IP* WHICH IS F(I)     GRA      1731     A
GRA                                                                              GRA      1732     A
GRA                SA1    NAB                                                    GRA      1733     A
GRA                ZR     X1,MTA12    IF NAB = 0                                 GRA      1734     A
GRA                SA0    4           TRM = 4       */ F(I), F(N) CASE           GRA      1735     A
GRA                                                                              GRA      1736     A
GRA      *         REMOVE VAR *UL* FROM CONSIDERATION                            GRA      1737     A
GRA                                                                              GRA      1738     A
GRA       MTA8     SA2    B7                                                     GRA      1739     A
GRA                SA3    B6+X2                                                  GRA      1740     A
GRA                SX6    X2                                                     GRA      1741     A
GRA                LX2    59-CA.TUP                                              GRA      1742     A
GRA                PL     X2,MTA11    IF ^TU[SI]    */ NOT VAR *UL*              GRA      1743     A
GRA                SA6    RAT         RAT = CO[SI]  */ SAVE ORD OF *UL*          GRA      1744     A
GRA                SB7    B7-B1       SI = SI - 1                                GRA      1745     A
GRA                MX1    1                                                      GRA      1746     A
GRA                LX1    1+CA.RAP                                               GRA      1747     A
GRA                BX7    X1+X3       RA[RCT(CO[SI])] = 1                        GRA      1748     A
GRA                SA7    A3                                                     GRA      1749     A
GRA                EQ     MTA11                                                  GRA      1750     A
GRA                                                                              GRA      1751     A
GRA      *         ASSIGN A REGISTER FOR THE *UL*                                GRA      1752     A
GRA                                                                              GRA      1753     A
GRA       MTA10    SA1    NAB                                                    GRA      1754     A
GRA                ZR     X1,MTA12    IF NAB = 0                                 GRA      1755     A
GRA                SX1    B1          INVF = 1                                   GRA      1756     A
GRA                RJ     AIR         ASSIGN IT TO A REGISTER                    GRA      1757     A
GRA                SX6    -B2                                                    GRA      1758     A
GRA                SA6    LUL+1       LUL(2) = REGNO                             GRA      1759     A
GRA                MX7    0                                                      GRA      1760     A
GRA                SA7    B6          [CO] = 0      */ IN CASE WE ASSIGNED 0     GRA      1761     A
GRA                SA7    B6+B1       [CO+1] = 0                                 GRA      1762     A
GRA                SA7    B6+2        [CO+2] = 0                                 GRA      1763     A
GRA                                                                              GRA      1764     A
GRA      *         ASSIGN REMAINING CANIDATES TO REGISTERS                       GRA      1765     A
GRA                                                                              GRA      1766     A
GRA       MTA11    SA5    B7                                                     GRA      1767     A
GRA                SA1    NAB                                                    GRA      1768     A
GRA                SB4    X5          I = CO[SI]                                 GRA      1769     A
GRA                ZR     X5,MTA14    IF I = 0      */ END OF TABLE              GRA      1770     A
GRA                ZR     X1,MTA14    IF NAB = 0    */ END OF REGS               GRA      1771     A
GRA                SA4    B6+B4                                                  GRA      1772     A
GRA                SX0    B1                                                     GRA      1773     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                LX4    -CA.DEFP                                          GRA        1774    A
GRA                BX1    -X4*X0       INVF = ^DEF[RCT(I)]                  GRA        1775    A
GRA                RJ     AIR          ASSIGN IT TO A REGISTER              GRA        1776    A
GRA                SX2    X7                                                GRA        1777    A
GRA                SB7    B7-B1        SI = SI - 1                          GRA        1778    A
GRA                NZ     X2,MTA11     IF UDT[CAW] " 0  */ PROGRAMMER VAR   GRA        1779    A
GRA                                                                        GRA        1780    A
GRA                SA4    A7+B1                                             GRA        1781    A
GRA                SA3    O.TET                                             GRA        1782    A
GRA                LX4    -IH.CAP      C = CA[RCT(I+1)]                     GRA        1783    A
GRA                SB3    X3                                                GRA        1784    A
GRA                SA2    B3+X4                                             GRA        1785    A
GRA                SX5    -B2                                               GRA        1786    A
GRA                LX5    T.REGP                                            GRA        1787    A
GRA                BX6    X5+X2        REG[TET(C)] = REGNO                  GRA        1788    A
GRA                SA6    A2                                                GRA        1789    A
GRA                EQ     MTA11                                             GRA        1790    A
GRA                                                                        GRA        1791    A
GRA        MTA12   SA0    0            TRM = 0                              GRA        1792    A
GRA                                                                        GRA        1793    A
GRA      *         SAVE FLAGS, CHECK TEST REPLACEMENT DECISION             GRA        1794    A
GRA                                                                        GRA        1795    A
GRA        MTA14   SA4    NTA                                               GRA        1796    A
GRA                SA5    NAB                                               GRA        1797    A
GRA                SA1    ABR                                               GRA        1798    A
GRA                SA2    MRA                                               GRA        1799    A
GRA                SX6    A0                                                GRA        1800    A
GRA                MX7    0                                                 GRA        1801    A
GRA                SA6    TRD          TRD = TRM                            GRA        1802    A
GRA                SA7    L.RND        L.RND = 0                            GRA        1803    A
GRA                IX6    X4-X5        NTA = NTA - NAB  */ N. TENATIVE ASSIGNMENTS  GRA  1804    A
GRA                SA6    A4                                                GRA        1805    A
GRA                MX7    -8                                                GRA        1806    A
GRA                BX3    X7+X1                                             GRA        1807    A
GRA                BX7    X3*X2        MRA = MRA & ( ABR + 77777400B )      GRA        1808    A
GRA                SA7    A2                                                GRA        1809    A
GRA                                                                        GRA        1810    A
GRA                SA1    LTT                                               GRA        1811    A
GRA                SX2    A0                                                GRA        1812    A
GRA                SX0    B1                                                GRA        1813    A
GRA                                                                        GRA        1814    A
GRA      *         TRM = 0 , SCAN *RCT* AND SET *EU* BIT FOR ALL CANIDATES THAT  GRA  1815    A
GRA      *         ARE MARKED AS *TU* SO *IRA* WILL MATERIALIZE THEM IN AN GRA  1816    A
GRA      *         X-REGISTER IN THE CASE THEY WERE ASSIGNED TO A B-REG.   GRA        1817    A
GRA                                                                        GRA        1818    A
GRA                SA3    L.RCT                                             GRA        1819    A
GRA                SB5    59-CA.TUP                                         GRA        1820    A
GRA                LX0    CA.EUP                                            GRA        1821    A
GRA                SB7    B6+X3        CI = RCT;  LIM = RCT + L.RCT         GRA        1822    A
GRA                SB4    B6                                                GRA        1823    A
GRA                ZR     X1,MTA17     IF LTT = 0    */ NO *TU* GAMES       GRA        1824    A
GRA                NZ     X2,MTA17     IF TRM " 0    */ *TU* BIT IS SUFFICIENT  GRA     1825    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                                                                    GRA     1826    A
GRA     MTA15   SA5     B6                                             GRA     1827    A
GRA             SB6     B6+3                                           GRA     1828    A
GRA             LX4     B5,X5                                          GRA     1829    A
GRA             PL      X4,MTA16    IF ^TU[CI]                         GRA     1830    A
GRA             BX6     X0+X5       EU[CI] = 1                         GRA     1831    A
GRA             SA6     A5                                             GRA     1832    A
GRA     MTA16   LT      B6,B7,MTA15 IF CI < LIM                        GRA     1833    A
GRA                                                                    GRA     1834    A
GRA     *       SCAN *RCT* FOR CANIDATES THAT ARE *IA* AND *RA* AND BUMP GRA   1835    A
GRA     *       *NOCC* OF THEIR INCREMENTS TO FORCE THEM TO A B-REGISTER. GRA  1836    A
GRA                                                                    GRA     1837    A
GRA     MTA17   SA1     MTAC                                           GRA     1838    A
GRA             SA2     A1+B1                                          GRA     1839    A
GRA             SA5     B7          CI = LIM                           GRA     1840    A
GRA             LX0     CA.NOCCP-CA.EUP                                GRA     1841    A
GRA             SB3     3                                              GRA     1842    A
GRA                                                                    GRA     1843    A
GRA     MTA18   SA5     A5-B3       CI = CI - 3                        GRA     1844    A
GRA             ZR      X5,MTA      IF [CI] = 0   */ END OF RCT        GRA     1845    A
GRA             BX6     -X1+X5                                         GRA     1846    A
GRA             NZ      X6,MTA18    IF ^(RA,VC,IA,RF)[CI]              GRA     1847    A
GRA             BX6     X2*X5                                          GRA     1848    A
GRA             NZ      X6,MTA18    IF KD[CI] ! ^RD[CI]                GRA     1849    A
GRA             SA3     A5+2        CCW = CI + 2                       GRA     1850    A
GRA             LX3     -CC.INCP                                       GRA     1851    A
GRA             SA4     B4+X3       CJ = RCT(INC[CCW])                 GRA     1852    A
GRA             IX6     X0+X4                                          GRA     1853    A
GRA             LX4     59-CA.CONP                                     GRA     1854    A
GRA             PL      X4,MTA18    IF ^CON[CJ]   */ INC IS NOT A *CON* GRA    1855    A
GRA             SA6     A4          NOCC[CJ] = NOCC[CJ] + 1            GRA     1856    A
GRA             EQ      MTA18                                          GRA     1857    A
GRA                                                                    GRA     1858    A
GRA      MTAC   BFMW    CA,(RA,VC,IA,RF)                               GRA     1859    A
GRA             BFMW    CA,(KD,^RD)                                    GRA     1860    A
GRA      AIR    TITLE   AIR - ASSIGN INDEX REGISTER                    GRA     1861    A
GRA     **      AIR - ASSIGN INDEX REGISTER                            GRA     1862    A
GRA     *                                                              GRA     1863    A
GRA     *       ENTRY   (B4) = I, RCT INDEX OF CANIDATE                GRA     1864    A
GRA     *               (B6) = [O.RCT]                                 GRA     1865    A
GRA     *               (X1) = INVF , = 1 IS INV IN LOOP, ELSE 0       GRA     1866    A
GRA     *                                                              GRA     1867    A
GRA     *       EXIT    (B2) = -REGNO ASSIGNED                         GRA     1868    A
GRA     *               (A7,A7) = RCT(I)                               GRA     1869    A
GRA     *               *RCT* ENTRY, RAT, NAB, ABR UPDATED             GRA     1870    A
GRA                                                                    GRA     1871    A
GRA      AIR1   SA2     A5+2        CCW = RCT(I+2)                     GRA     1872    A
GRA             LX5     1+CA.RAP                                       GRA     1873    A
GRA             BX7     X5                                             GRA     1874    A
GRA             MX6     -CC.REG1L                                      GRA     1875    A
GRA             LX2     -CC.REG1P                                      GRA     1876    A
GRA             BX4     -X2+X6                                         GRA     1877    A

        0         1         2         3         4         5         6         7         8
        12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SA7    A5+                                                    GRA        1878     A
GRA                SB2    X4          REGNO = REG1[CCW]                          GRA        1879     A
GRA                                                                              GRA        1880     A
GRA       AIR      ROUTINE                                                        GRA        1881     A
GRA                SA5    B6+B4       CAW = RCT(I)                               GRA        1882     A
GRA                LX5    59-CA.RAP                                              GRA        1883     A
GRA                MI     X5,AIR1     IF RA[CAW]    */ REG PREVIOUSLY ASSIGNED   GRA        1884     A
GRA                                                                              GRA        1885     A
GRA                SX2    B1                                                     GRA        1886     A
GRA                SA4    ABR                                                    GRA        1887     A
GRA                SA3    NAB                                                    GRA        1888     A
GRA                NX7    B2,X4                                                  GRA        1889     A
GRA                IX6    X3-X2       NAB = NAB - 1                              GRA        1890     A
GRA                SB2    B2-47       REGNO = 47 - NORMC( ABR )                  GRA        1891     A
GRA                SA6    A3                                                     GRA        1892     A
GRA                LX2    -B2,X2                                                 GRA        1893     A
GRA                BX7    -X2*X4      ABR = ABR & ^SHIFT(1,REGNO)                GRA        1894     A
GRA                SA7    A4                                                     GRA        1895     A
GRA                SX6    -B2                                                    GRA        1896     A
GRA                SX7    B4                                                     GRA        1897     A
GRA                LX1    RA.INVP                                                GRA        1898     A
GRA                BX7    X1+X7       INV[RAT(REGNO)] = INVF                     GRA        1899     A
GRA                SA7    RAT+X6      R1[RAT(REGNO)] = I  */ RAT POINTS TO RCT   GRA        1900     A
GRA                LX6    CC.REG1P                                               GRA        1901     A
GRA                MX4    1                                                      GRA        1902     A
GRA                BX7    X4+X5       RA[CAW] = 1                                GRA        1903     A
GRA                LX7    1+CA.RAP                                               GRA        1904     A
GRA                SA7    A5                                                     GRA        1905     A
GRA                SA3    A5+2                                                   GRA        1906     A
GRA                BX6    X6+X3       REG1[RCT(I+2)] = REGNO                     GRA        1907     A
GRA                SA6    A3                                                     GRA        1908     A
GRA                EQ     AIR                                                    GRA        1909     A
GRA        DXA     TITLE  DXA - DETERMINE X-ASSIGNMENTS                          GRA        1910     A
GRA       **       DXA - DETERMINE X-ASSIGNMENTS IN AN INNERMOST LOOP            GRA        1911     A
GRA       *        FORM X-REGISTER CANIDATE TABLE AND SORT.                      GRA        1912     A
GRA       *        SELECT A MAX OF 4 CANIDATES AND SEE IF THE LOOP CAN BE CODED  GRA        1913     A
GRA       *        WITH THEM LOCKED IN REGISTERS.  COUNT NUMBER OF REMAINING     GRA        1914     A
GRA       *        MEMORY REFS IN THE LOOP AND MAKE THE X-ASSIGNMENTS.           GRA        1915     A
GRA       *                                                                      GRA        1916     A
GRA       *        ENTRY  (XRF) = 0 , NO EXTERNAL REFS                           GRA        1917     A
GRA       *               MAXW < 8 + NTA , X-REG ASSIGNMENT MAY BE POSSIBLE      GRA        1918     A
GRA       *                                                                      GRA        1919     A
GRA       *        EXIT   (MLW) = MAX REG WIDTH IN LOOP BODY AFTER X-ASSIGNMENT  GRA        1920     A
GRA       *               (NXC) = NUMBER OF X-CANIDATES ASSIGNED                 GRA        1921     A
GRA                                                                              GRA        1922     A
GRA        DXAA    BFMW   CA,(VC,CON)                                            GRA        1923     A
GRA                BFMW   CA,(RA,RF)                                             GRA        1924     A
GRA                                                                              GRA        1925     A
GRA        DXA     ROUTINE                                                       GRA        1926     A
GRA                SA5    NXC                                                    GRA        1927     A
GRA                NZ     X5,DXA4A    IF NXC " 0    */ X-REGS ASSIGNED IN *ERC*  GRA        1928     A
GRA                SA4    N.HB                                                   GRA        1929     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                NZ     X4,DXA      IF N.HB " 0   */ NOT AN INNERMOST LOOP          GRA        1930     A
GRA                                                                                   GRA        1931     A
GRA                SA1    NVC                                                         GRA        1932     A
GRA                ALLOC  RND,X1      ALLOC( RND , NVC )                              GRA        1933     A
GRA                SA4    O.RCT                                                       GRA        1934     A
GRA                SA5    L.RCT                                                       GRA        1935     A
GRA                MX7    0                                                           GRA        1936     A
GRA                SA7    X2          SI = [O.RCT]; [SI] = 0                          GRA        1937     A
GRA                MX0    -CA.SORTL                                                   GRA        1938     A
GRA                SB7    X2+B1       SO = SI + 1                                     GRA        1939     A
GRA                LX0    CA.SORTP                                                    GRA        1940     A
GRA                SB6    X4          CO = [O.RCT]                                    GRA        1941     A
GRA                SA1    DXAA                                                        GRA        1942     A
GRA                SA2    A1+B1                                                       GRA        1943     A
GRA                SA5    B6+X5       CI = CO + L.RCT                                 GRA        1944     A
GRA                SB3    3                                                           GRA        1945     A
GRA                                                                                   GRA        1946     A
GRA      *         FORM X-CANIDATE TABLE AND SORT                                     GRA        1947     A
GRA                                                                                   GRA        1948     A
GRA       DXA1     SA5    A5-B3       CI = CI - 3                                     GRA        1949     A
GRA                ZR     X5,DXA2     IF [CI] = 0   */ END OF TABLE                   GRA        1950     A
GRA                BX6    X1*X5                                                       GRA        1951     A
GRA                ZR     X6,DXA1     IF ^( VC[CI] ! CON[CI] )                        GRA        1952     A
GRA                BX3    X2*X5                                                       GRA        1953     A
GRA                NZ     X3,DXA1     IF RA[CI] ! RF[CI]                              GRA        1954     A
GRA                                                                                   GRA        1955     A
GRA                BX3    -X0*X5                                                      GRA        1956     A
GRA                SX4    A5-B6       I = CI - CO                                     GRA        1957     A
GRA                IX7    X3+X4                                                       GRA        1958     A
GRA                SA7    A7+B1       SI = SI + 1;  [SI] = SORT[CI] ! I               GRA        1959     A
GRA                EQ     DXA1                                                        GRA        1960     A
GRA                                                                                   GRA        1961     A
GRA       DXA2     SB3    A7+B1       SI = SI + 1   */ LWA + 1                        GRA        1962     A
GRA                MX7    0                                                           GRA        1963     A
GRA                SX1    B3-B7       LEN = SI - SO                                   GRA        1964     A
GRA                SA7    L.RND       L.RND = 0                                       GRA        1965     A
GRA                SX6    DXA                                                         GRA        1966     A
GRA                SA6    CRWA        [CRWA] = DXA  */ FAIL EXIT ADDR                 GRA        1967     A
GRA                ZR     X1,DXA5     IF LEN = 0    */ GO COMPUTE MLW, N.LD, N.ST     GRA        1968     A
GRA                                                                                   GRA        1969     A
GRA                CALL   SHL         SORT( RND )                                     GRA        1970          I
          -CCG1
GRA                CALL   SHL=        SORT( RND )                                     CCG1         95          I
          -CCGNC
GRA                CALL   SHL#        SORT( RND )                                     CCGNC       650     A
GRA                                                                                   GRA        1971     A
GRA      *         MOVE CANIDATES WITH HIGHEST PRIORITY TO *XCT*                      GRA        1972     A
GRA      *         AND SET *RA* BIT FOR WINNERS IN *RCT*.                             GRA        1973     A
GRA                                                                                   GRA        1974     A
GRA                SB4    XCT         I = 0                                           GRA        1975     A
GRA                SB5    B4+4        LIM = 4                                         GRA        1976     A
GRA                MX0    1                                                           GRA        1977     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                LX0    1+CA.RAP                                                GRA        1978     A
GRA                SB2    B4+                                                     GRA        1979     A
GRA                                                                               GRA        1980     A
GRA        DXA3    SA1    B3-B1        SW = [SI-1]                                GRA        1981     A
GRA                SA2    B6+X1                                                   GRA        1982     A
GRA                SX6    X1           J = XCT(I) = CO[SW]                        GRA        1983     A
GRA                BX7    X0+X2        RA[RCT(J)] = 1                             GRA        1984     A
GRA                SA6    B4                                                      GRA        1985     A
GRA                SB4    B4+B1        I = I + 1                                  GRA        1986     A
GRA                SA7    A2                                                      GRA        1987     A
GRA                SB3    B3-B1        SI = SI - 1                                GRA        1988     A
GRA                EQ     B4,B5,DXA4   IF I = LIM    */ LIM CANIDATES COLLECTED   GRA        1989     A
GRA                GT     B3,B7,DXA3   IF SI > SO    */ NOT END OF SORT TABLE     GRA        1990     A
GRA                                                                               GRA        1991     A
GRA        *       COMPUTE REG WIDTH OF LOOP, IF FAILURE, REDUCE NUMBER OF        GRA        1992     A
GRA        *       X-CANIDATES, ELSE MAKE FINAL ASSIGNMENTS                       GRA        1993     A
GRA                                                                               GRA        1994     A
GRA        DXA4    SX6    B4-B2                                                   GRA        1995     A
GRA                SA6    NXC          NXC = I       */ N. X-CANIDATES            GRA        1996     A
GRA                                                                               GRA        1997     A
GRA        DXA4A   SX6    DXA6                                                    GRA        1998     A
GRA                SA6    CRWA         [CRWA] = DXA6 */ FAIL EXIT ADDRESS          GRA        1999     A
GRA                                                                               GRA        2000     A
GRA        DXA5    SX7    0                                                       GRA        2001     A
GRA                SA7    MLW          MLW = 0       */ MAX REG WITH OF LOOP      GRA        2002     A
GRA                                                                               GRA        2003     A
GRA                CLCM   CRW          COMPUTE REG WIDTH                          GRA        2004     A
GRA                SA1    N.HB                                                    GRA        2005     A
GRA                ZR     X1,DXA7      IF N.HB = 0   */ INNERMOST LOOP            GRA        2006     A
GRA                                                                               GRA        2007     A
GRA                EQ     DXA                                                     GRA        2008     A
GRA                                                                               GRA        2009     A
GRA        *       CRW FAIL EXIT, REDUCE X-REG CANIDATE COUNT                     GRA        2010     A
GRA                                                                               GRA        2011     A
GRA        DXA6    SA1    NXC                                                     GRA        2012     A
GRA                SA2    XCT-1+X1     I = XCT(NXC)                               GRA        2013          I
           -FCC0811
GRA                MX0    2                                                       GRA        2014          I
           -FCC0811
GRA                SA3    B6+X2                                                   GRA        2015          I
           -FCC0811
GRA                LX0    2+CA.RAFP                                               GRA        2016          I
           -FCC0811
GRA                SX6    X1-1                                                    GRA        2017          I
           -FCC0811
GRA                SA6    A1                                                      GRA        2018          I
           -FCC0811
GRA                BX7    -X0*X3       (RA,IRA)RCT(J) = 0  */ CLEAR ASSIGNMENT    GRA        2019          I
           -FCC0811
GRA                SA7    A3                                                      GRA        2020          I
           -FCC0811
GRA                RJ     RXA          REMOVE LAST X-ASSIGNMENT                   FCC0811       8     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                NZ     X6,DXA5     IF NXC " 0    */ MORE CANIDATES LEFT          GRA    2021    A
GRA                                                                                  GRA    2022    A
GRA                EQ     DXA                                                        GRA    2023    A
GRA        DXA     SPACE  3,14                                                       GRA    2024    A
GRA      *         SUCCESS, ASSIGN CANIDATES TO REGISTERS                            GRA    2025    A
GRA      *         CHECK NUMBER OF REMAINING STORES AND REMOVE X6,X7 FROM            GRA    2026    A
GRA      *         CONSIDERATION IF THEIR ARE TOO MANY LEFT.                         GRA    2027    A
GRA                                                                                  GRA    2028    A
GRA        DXA7    CLCM   CMR         COUNT NUMBER OF MEMORY REFS LEFT IN LOOP      GRA    2029    A
GRA                                                                                  GRA    2030    A
GRA                SA5    NXC                                                        GRA    2031    A
GRA                ZR     X5,DXA      IF NXC = 0    */ NO X-CANIDATES               GRA    2032    A
GRA                                                                                  GRA    2033    A
GRA                SA1    MRA                                                        GRA    2034    A
GRA                SA2    N.ST                                                       GRA    2035    A
GRA                LX1    -16         AXR = SHIFT(MRA,-16)                           GRA    2036    A
GRA                MX0    0           STL = 0                                        GRA    2037    A
GRA                ZR     X2,DXA8     IF N.ST = 0   */ NO STORES LEFT IN THE LOOP    GRA    2038    A
GRA                SX0    1S6         STL = 1S6     */ LOCKOUT X6                    GRA    2039    A
GRA                AX3    B1,X2                                                      GRA    2040    A
GRA                ZR     X3,DXA8     IF N.ST = 1   */ ONLY 1 ST PER BLOCK           GRA    2041    A
GRA                SX0    3S6         STL = 3S6     */ LOCKOUT THE STORE REGS        GRA    2042    A
GRA                                                                                  GRA    2043    A
GRA      *         PICK THE SET OF REGISTERS USED FOR THE *LD* ONLY CANIDATES        GRA    2044    A
GRA      *         SO THAT SUFFICIENT LD REGS WILL BE AVAIL FOR THE OTHER LD"S.      GRA    2045    A
GRA                                                                                  GRA    2046    A
GRA        DXA8    SA5    N.LD                                                       GRA    2047    A
GRA                SA0    321670B     LRS = 076123  */ LD ONLY REG SET               GRA    2048    A
GRA                SB2    X5-3                                                       GRA    2049    A
GRA                GT     B2,DXA9     IF N.LD > 3                                    GRA    2050    A
GRA                SA0    367021B     LRS = 120763                                   GRA    2051    A
GRA                ZR     B2,DXA9     IF N.LD = 3                                    GRA    2052    A
GRA                SA0    670321B     LRS = 123076                                   GRA    2053    A
GRA                                                                                  GRA    2054    A
GRA        DXA9    SA2    NXC                                                        GRA    2055    A
GRA                SA3    O.RCT                                                      GRA    2056    A
GRA                BX1    -X0*X1      AXR = AXR & ^STL                               GRA    2057    A
GRA                SB4    XCT         I = 0                                          GRA    2058    A
GRA                SB5    B4+X2       L = NXC                                        GRA    2059    A
GRA                MX2    -3                                                         GRA    2060    A
GRA                SB6    X3                                                         GRA    2061    A
GRA                SX0    1                                                          GRA    2062    A
GRA                                                                                  GRA    2063    A
GRA      *         PICK X-REGISTERS FOR THE CANIDATES AS A FUNCTION OF THEIR         GRA    2064    A
GRA      *         TYPE AND USEAGE.                                                  GRA    2065    A
GRA                                                                                  GRA    2066    A
GRA        DXA10   SA4    B4          J = XCT(I)                                     GRA    2067    A
GRA                SA5    B6+X4       CAW = RCT(J)                                   GRA    2068    A
GRA                SX7    321670B     XRS = 076123  */ X-REG SET IN SEARCH ORDER     GRA    2069    A
GRA                SB2    X5                                                         GRA    2070    A
GRA                LX5    59-CA.CONP                                                 GRA    2071    A
GRA                MI     X5,DXA11    IF CON[CAW]                                    GRA    2072    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                ZR     B2,DXA11    IF UDI[CAW] = 0  */ COMPILER TEMP                GRA     2073    A
GRA                                                                                    GRA     2074    A
GRA                LX5    CA.CONP-CA.DEFP                                              GRA     2075    A
GRA                SX7    A0          XRS = LRS                                        GRA     2076    A
GRA                PL     X5,DXA11    IF ^DEF[CAW]  */ LOAD ONLY                       GRA     2077    A
GRA                SX7    321067B     XRS = 760123                                     GRA     2078    A
GRA                                                                                    GRA     2079    A
GRA       DXA11    BX6    -X2*X7      REGN = XRS & 7                                   GRA     2080    A
GRA                AX7    3           XRS = SHIFT(XRS,-3)                              GRA     2081    A
GRA                SB2    X6                                                           GRA     2082    A
GRA                LX3    B2,X0       RBIT = SHIFT(1,REGN)                             GRA     2083    A
GRA                BX5    X3*X1                                                        GRA     2084    A
GRA                ZR     X5,DXA11    IF AXR & RBIT = 0 */ REG NOT AVAILABLE           GRA     2085    A
GRA                                                                                    GRA     2086    A
GRA                SA5    A5                                                           GRA     2087    A
GRA                BX1    -X3*X1      AXR = AXR & ^RBIT  */ REMOVE FROM AVAIL SET      GRA     2088    A
GRA                SX6    X6+20B      REGN = REGN + 20B  */ X-REG NUMBER               GRA     2089    A
GRA                SX4    X4          R1[RAT(REGNO)] = J                               GRA     2090    A
GRA                LX5    -CA.DEFP                                                     GRA     2091    A
GRA                BX3    -X5*X0      INV[RAT(REGNO)] = ^DEF[RCT(J)]                   GRA     2092    A
GRA                SA5    A5+2                                                         GRA     2093    A
GRA                LX3    RA.INVP                                                      GRA     2094    A
GRA                BX7    X3+X4                                                        GRA     2095    A
GRA                SA7    RAT+X6                                                       GRA     2096    A
GRA                LX6    CC.REG1P                                                     GRA     2097    A
GRA                BX6    X6+X5       REG1[RCT(J+2)] = REGN                            GRA     2098    A
GRA                SB4    B4+1        I = I + 1                                        GRA     2099    A
GRA                SA6    A5                                                           GRA     2100    A
GRA                LT     B4,B5,DXA10 IF I < NXC                                       GRA     2101    A
GRA                                                                                    GRA     2102    A
GRA                LX1    16                                                           GRA     2103    A
GRA                BX6    X1                                                           GRA     2104    A
GRA                SA6    MRA         MRA = SHIFT(AXR,16)                              GRA     2105    A
GRA                EQ     DXA                                                          GRA     2106    A
GRA        CRW     TITLE  CRW - COMPUTE REGISTER WIDTH                                 GRA     2107    A
GRA       **       CRW - COMPUTE REGISTER WIDTH OF A SEQUENCE                          GRA     2108    A
GRA       *                                                                            GRA     2109    A
GRA       *        LINK WORDS CONTAIN *RCT* ORDINALS OF CANIDATES AND *RA* BIT         GRA     2110    A
GRA       *        IS SET IF IT IS ASSIGNED TO A REGISTER.                             GRA     2111    A
GRA       *        CANIDATES PLACED IN LOCKED REGISTERS ARE ASSUMED TO BE *LIVE*       GRA     2112    A
GRA       *        THORUGHOUT THE LOOP.                                                GRA     2113    A
GRA       *                                                                            GRA     2114    A
GRA       *        ENTRY  (X0) = R-NUMBER MASK                                         GRA     2115    A
GRA       *               (X5) = O.SEQ                                                 GRA     2116    A
GRA       *               (NXC) = NUMBER OF LOCKED X-REGISTERS                         GRA     2117    A
GRA       *                                                                            GRA     2118    A
GRA       *        EXIT   (B6) = [O.RCT]                                               GRA     2119    A
GRA       *               (MLW) = MAX REG WIDTH OF LOOP                                GRA     2120    A
GRA                                                                                    GRA     2121    A
GRA                DESCRIBE I.,60     LINK WORD                                        GRA     2122    A
GRA                DEFINE 24                                                           GRA     2123    A
GRA        USES    DEFINE 18          REMAINING USES OF NON CANIDATES                  GRA     2124    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA        CO       DEFINE 18          CANIDATE ORDINAL ( *RCT* )                    GRA        2125     A
GRA                                                                                   GRA        2126     A
GRA        CRWA     CON    377777B     FAIL EXIT ADDRESS PROCESSOR                   GRA        2127     A
GRA        ASPA     BFMW   D,(BM,ST)                                                 GRA        2128     A
GRA                                                                                   GRA        2129     A
GRA        CRW      ROUTINE                                                          GRA        2130     A
GRA                 SA1    L.SEQ                                                     GRA        2131     A
GRA                 SB2    X1-8                                                      GRA        2132     A
GRA                 LE     B2,CRW      IF L.SEQ @ 8  */ EMPTY BLOCK                  GRA        2133     A
GRA                                                                                   GRA        2134     A
GRA                 RJ     SLW         SAVE LINK WORDS                               GRA        2135     A
GRA                 SA3    O.RCT                                                     GRA        2136     A
GRA                 S"TB"  X5+3        TB = O.SEQ + 3                                GRA        2137     A
GRA                 SB6    X3                                                        GRA        2138     A
GRA                                                                                   GRA        2139     A
GRA        *        FIRST ADJUST THE PRED OF STORES THAT ARE LOCKED IN REGISTERS      GRA        2140     A
GRA                                                                                   GRA        2141     A
GRA                 SA1    ASPA                                                      GRA        2142     A
GRA                 SA4    "TB"-B1     DI = TB - 1                                   GRA        2143     A
GRA                 SB4    CA.RAP-CA.LXP                                             GRA        2144     A
GRA                 EQ     ASP2                                                      GRA        2145     A
GRA                                                                                   GRA        2146     A
GRA        ASP1     UX6    B2,X5                                                     GRA        2147     A
GRA                 ZR     B2,CRW0     IF OC[R1] = OC.EOQ                            GRA        2148     A
GRA                                                                                   GRA        2149     A
GRA        ASP2     SA4    A4+4        DI = DI + 4                                   GRA        2150     A
GRA                 BX6    X1*X4                                                     GRA        2151     A
GRA                 LX4    59-D.BMP                                                  GRA        2152     A
GRA                 ZR     X6,ASP2     IF ^( BM[DI] ! ST[DI] )                       GRA        2153     A
GRA                                                                                   GRA        2154     A
GRA                 SA5    A4-2        R1 = DI - 2                                   GRA        2155     A
GRA                 MI     X4,ASP1     IF BM[DI]                                     GRA        2156     A
GRA                                                                                   GRA        2157     A
GRA                 SA2    A4+B1       LI = DI + 1;  I = CO[LI]                      GRA        2158     A
GRA                 SA3    B6+X2       CI = RCT(I)                                   GRA        2159     A
GRA                 LX3    59-CA.RAP                                                 GRA        2160     A
GRA                 PL     X3,ASP2     IF ^RA[CI]                                    GRA        2161     A
GRA                                                                                   GRA        2162     A
GRA                 LX6    B4,X3                                                     GRA        2163     A
GRA                 LX3    CA.RAP-CA.PSPP                                            GRA        2164     A
GRA                 PL     X6,ASP3     IF ^LX[CI]    */ DEAD ON EXIT                 GRA        2165     A
GRA                 PL     X3,ASP2     IF ^PSP[CI]   */ STORE CANT MOVE              GRA        2166     A
GRA                                                                                   GRA        2167     A
GRA        ASP3     BX5    "RN"X5      R = RI[R1]                                    GRA        2168     A
GRA                 SX7    X2+                                                       GRA        2169     A
GRA                 SA7    "TB"+X5     LI = TB + R;  [LI] = I                        GRA        2170     A
GRA                 EQ     ASP2                                                      GRA        2171     A
GRA                                                                                   GRA        2172     A
GRA        *        INITIALIZE FOR THE REG WIDTH CALCULATION                          GRA        2173     A
GRA                                                                                   GRA        2174     A
GRA        CRW0     SA2    NXC                                                       GRA        2175     A
GRA                 SB4    X2          RW = NXC      */ REG WIDTH                    GRA        2176     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA              SA0    X2                                                GRA      2177    A
GRA              SB5    X2          MW = NXC      */ MAX WIDTH            GRA      2178    A
GRA              MX2    -D.USESL                                          GRA      2179    A
GRA              SX1    B1                                                GRA      2180    A
GRA              LX2    I.USESP                                           GRA      2181    A
GRA              SA4    "TB"-1      DI = TB - 1                           GRA      2182    A
GRA              SB3    OC.RS                                             GRA      2183    A
GRA                                                                       GRA      2184    A
GRA      *       ADVANCE TO NEXT INSTRUCTION, JUMP ON TYPE                GRA      2185    A
GRA                                                                       GRA      2186    A
GRA       CRW1   SA5    A4+2        R1 = DI + 2                           GRA      2187    A
GRA              SA4    A5+2        DI = R1 + 2                           GRA      2188    A
GRA              UX5    B2,X5                                             GRA      2189    A
GRA              LX4    58-D.TYP                                          GRA      2190    A
GRA              IX7    X4+X4                                             GRA      2191    A
GRA              MI     X4,CRW3     IF TYPE[DI] = III ! IV                GRA      2192    A
GRA              MI     X7,CRW2     IF TYPE[DI] = II                      GRA      2193    A
GRA                                                                       GRA      2194    A
GRA      *       TYPE I                                                   GRA      2195    A
GRA                                                                       GRA      2196    A
GRA              LX4    1+D.TYP-D.^DP     RNUMS = RJRKRI[R1]              GRA      2197    A
GRA              MI     X4,CRW7     IF ^D[DI]                             GRA      2198    A
GRA              AX5    R1.RIL      RNUMS = RJRK[R1]                      GRA      2199    A
GRA              EQ     CRW7                                              GRA      2200    A
GRA                                                                       GRA      2201    A
GRA      *       TYPE II                                                  GRA      2202    A
GRA                                                                       GRA      2203    A
GRA       CRW2   LX4    1+D.TYP-D.^DP                                     GRA      2204    A
GRA              NE     B2,B3,CRW9  IF OC[R1] " OC.RS                     GRA      2205    A
GRA              EQ     CRW1                                              GRA      2206    A
GRA                                                                       GRA      2207    A
GRA      *       TYPE IV                                                  GRA      2208    A
GRA                                                                       GRA      2209    A
GRA       CRW3   PL     X7,CRW4     IF TYPE[DI] = III                     GRA      2210    A
GRA              SB4    A0+         RW = NXC                              GRA      2211    A
GRA              NZ     B2,CRW1     IF OC[R1] " OC.EOQ                    GRA      2212    A
GRA                                                                       GRA      2213    A
GRA              RJ     RLW         RESTORE LINK WORDS                    GRA      2214    A
GRA              SA1    MLW                                               GRA      2215    A
GRA              SX6    B5                                                GRA      2216    A
GRA              SB3    X1                                                GRA      2217    A
GRA              GE     B3,B5,CRW   MLW = MAX( MLW , MW )                 GRA      2218    A
GRA              SA6    A1+                                               GRA      2219    A
GRA              EQ     CRW                                               GRA      2220    A
GRA                                                                       GRA      2221    A
GRA      *       TYPE III                                                 GRA      2222    A
GRA                                                                       GRA      2223    A
GRA       CRW4   LX4    1+D.TYP-D.^DP     RNUMS = RI[R1]                  GRA      2224    A
GRA              MI     X4,CRW5     IF ^D[DI]                             GRA      2225    A
GRA              MX5    0           RNUMS = 0                             GRA      2226    A
GRA       CRW5   SA3    A5+B1       R2 = R1 + 1                           GRA      2227    A
GRA              BX5    "RN"X5                                            GRA      2228    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                LX3    -IH.RFP                                          GRA        2229     A
GRA                BX6    "RN"X3       R = RF[R2]                          GRA        2230     A
GRA                ZR     X6,CRW7      IF R = 0                            GRA        2231     A
GRA                LX5    R1.RIL                                           GRA        2232     A
GRA                BX5    X6+X5        RNUMS = SHIFT(RNUMS,R1.RIL) ! R     GRA        2233     A
GRA                                                                       GRA        2234     A
GRA       *        DECREMENT USES OF OPERANDS ( RNUMS )                    GRA        2235     A
GRA                                                                       GRA        2236     A
GRA        CRW7    BX7    "RN"X5       R = RI[RNUMS]                       GRA        2237     A
GRA                AX5    R1.RIL       RNUMS = SHIFT(RNUMS,-R1.RIL)        GRA        2238     A
GRA                ZR     X7,CRW8      IF R = 0                            GRA        2239     A
GRA                                                                       GRA        2240     A
GRA                SA3    "TB"+X7      LI = TB + R                         GRA        2241     A
GRA                LX3    -I.USESP                                         GRA        2242     A
GRA                SB2    X3+          U = USES[LI]                        GRA        2243     A
GRA                ZR     B2,CRW8      IF U = 0      */ LOCKED IN A REGISTER GRA      2244     A
GRA                IX6    X3-X1        USES[LI] = USES[LI] - 1             GRA        2245     A
GRA                LX6    I.USESP                                          GRA        2246     A
GRA                SA6    A3                                               GRA        2247     A
GRA                GT     B2,B1,CRW8   IF U > 1      */ NOT LAST USE       GRA        2248     A
GRA                SB4    B4-B1        RW = RW - 1                         GRA        2249     A
GRA        CRW8    NZ     X5,CRW7      IF RNUMS " 0                        GRA        2250     A
GRA                                                                       GRA        2251     A
GRA       *        ADVANCE *RW* TO ACCOUNT FOR THE NEW RESULT              GRA        2252     A
GRA                                                                       GRA        2253     A
GRA        CRW9    SA3    A4+B1        LI = DI + 1                         GRA        2254     A
GRA                MI     X4,CRW1      IF ^D[DI]     */ ^DEF A NEW RESULT  GRA        2255     A
GRA                SA5    B6+X3                                            GRA        2256     A
GRA                LX5    59-CA.RAP                                        GRA        2257     A
GRA                MI     X5,CRW1      IF RA[RCT(CO[LI])]  */ IN A LOCKED REG GRA      2258     A
GRA                                                                       GRA        2259     A
GRA                LX4    D.^DP-D.PRSP                                     GRA        2260     A
GRA                PL     X4,CRW10     IF ^PRS[CI]                         GRA        2261     A
GRA                                                                       GRA        2262     A
GRA                SA5    A3+B1        R1RS = LI + 1  */ R1 OF FOLLOWING RS GRA        2263     A
GRA                LX5    58-R1.SOP-SO.RTP                                 GRA        2264     A
GRA                PL     X5,CRW1      IF SORT[R1RS] " 2  */ NOT TO AN X-REGISTER GRA  2265     A
GRA                                                                       GRA        2266     A
GRA        CRW10   LX4    1+D.PRSP-D.USESP+I.USESP                         GRA        2267     A
GRA                BX5    -X2*X4                                           GRA        2268     A
GRA                SB4    B4+B1        RW = RW + 1                         GRA        2269     A
GRA                BX6    X5+X3        USES[LI] = USES[DI]                 GRA        2270     A
GRA                SA6    A3                                               GRA        2271     A
GRA                LE     B4,B5,CRW1   IF RW @ MW                          GRA        2272     A
GRA                                                                       GRA        2273     A
GRA                SB5    B4           MW = RW       */ SET NEW MAX       GRA        2274     A
GRA                SB2    9                                                GRA        2275     A
GRA                LT     B5,B2,CRW1   IF MW < 9     */ HAVENT JAMMED THE REGS GRA   2276     A
GRA                                                                       GRA        2277     A
GRA                RJ     RLW          RESTORE LINK WORDS                  GRA        2278     A
GRA                SA1    CRWA                                             GRA        2279     A
GRA                SX6    B2                                               GRA        2280     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SB3    X1                                                    GRA        2281     A
GRA                SA6    MLW         MLW = 9                                   GRA        2282     A
GRA                JP     B3          JUMP( [CRWA] )  */ EXIT TO FAIL ADDRESS   GRA        2283     A
GRA        SLW     SPACE  3,14        SLW                                       GRA        2284     A
GRA       **       SLW - SAVE LINK WORDS                                        GRA        2285     A
GRA       *                                                                     GRA        2286     A
GRA       *        ENTRY  (X1) = L.SEQ                                           GRA        2287     A
GRA       *                                                                     GRA        2288     A
GRA       *        EXIT   (X5) = [O.SEQ]                                         GRA        2289     A
GRA                                                                             GRA        2290     A
GRA        SLW     ROUTINE                                                       GRA        2291     A
GRA                AX1    2                                                     GRA        2292     A
GRA                SB2    X1-2                                                  FCC2051       6     A
GRA                ZR     B2,SLW      IF L.SEQ = 8  */ EMPTY BLOCK              FCC2051       7     A
GRA                ALLOC  RND,X1      ALLOC( RND , L.SEQ/4 )                    GRA        2293     A
GRA                SA5    O.SEQ                                                 GRA        2294     A
GRA                SB2    X1-2        I = L.SEQ/4 - 2                           GRA        2295     A
GRA                SA4    X5+7        SI = O.SEQ + 7                            GRA        2296     A
GRA                BX6    X4                                                    GRA        2297     A
GRA                SA6    X2          RI = [O.RND];  [RI] = [SI]                GRA        2298     A
GRA                                                                             GRA        2299     A
GRA        SLW1    SA4    A4+4        SI = SI + 4                               GRA        2300     A
GRA                SB2    B2-B1       I = I - 1                                 GRA        2301     A
GRA                BX6    X4                                                    GRA        2302     A
GRA                SA6    A6+B1       RI = RI + 1;  [RI] = [SI]                 GRA        2303     A
GRA                NZ     B2,SLW1     IF I " 0                                  GRA        2304     A
GRA                                                                             GRA        2305     A
GRA                EQ     SLW                                                   GRA        2306     A
GRA        RLW     SPACE  2           RLW                                       GRA        2307     A
GRA       **       RLW - RESTORE LINK WORDS                                     GRA        2308     A
GRA       *                                                                     GRA        2309     A
GRA       *        PRESERVES  B - 2, 3, 5, 6, 7                                 GRA        2310     A
GRA                                                                             GRA        2311     A
GRA        RLW     ROUTINE                                                       GRA        2312     A
GRA                SA1    O.RND                                                 GRA        2313     A
GRA                SA2    L.SEQ                                                 GRA        2314     A
GRA                SB4    4                                                     GRA        2315     A
GRA                SA3    X1          RI = [O.RND]                              GRA        2316     A
GRA                AX2    2                                                     GRA        2317     A
GRA                SX7    X2-2        I = L.SEQ/4 - 2                           GRA        2318     A
GRA                ZR     X7,RLW      IF I = 0                                  FCC2051       8     A
GRA                SA5    O.SEQ                                                 GRA        2319     A
GRA                BX6    X3                                                    GRA        2320     A
GRA                SA6    X5+7        SI = O.SEQ + 7;  [SI] = [RI]              GRA        2321     A
GRA                SX2    B1                                                    GRA        2322     A
GRA                                                                             GRA        2323     A
GRA        RLW1    SA3    A3+B1       RI = RI + 1                               GRA        2324     A
GRA                IX7    X7-X2       I = I - 1                                 GRA        2325     A
GRA                BX6    X3                                                    GRA        2326     A
GRA                SA6    A6+B4       SI = SI + 4;  [SI] = [RI]                 GRA        2327     A
GRA                NZ     X7,RLW1     IF I " 0                                  GRA        2328     A
GRA                                                                             GRA        2329     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SA7    L.RND       L.RND = 0                                  GRA        2330     A
GRA                EQ     RLW                                                    GRA        2331     A
GRA       CMR      TITLE  CMR - COUNT MEMORY REFERENCES                          GRA        2332     A
GRA      **        CMR - COUNT REMAINING MEMORY REFERENCES IN A BLOCK            GRA        2333     A
GRA      *         THE PURPOSE OF THIS ROUTINE IS TO SEE HOW MANY LOADS AND      GRA        2334     A
GRA      *         STORES ARE LEFT IN THE LOOP SO WE CAN MAKE AN INTELLIGENT     GRA        2335     A
GRA      *         DECISION ABOUT ASSIGNING STORE REGISTERS IN *DXA* AND LOAD    GRA        2336     A
GRA      *         REGISTERS IN *DAA* WHEN TRYING TO SETUP PREFETCH CODE.        GRA        2337     A
GRA      *         IN THE CASE THAT A STORE CANNOT MOVE OUT OF A LOOP THIS       GRA        2338     A
GRA      *         SCHEME BACKFIRES AND THE CANIDATE WILL PROBABELY GET ASSIGNED GRA        2339     A
GRA      *         TO A LOAD REGISTER, FORCEING TRANSMITS IN AND OUT OF THE LOOP. GRA       2340     A
GRA      *                                                                       GRA        2341     A
GRA      *         ENTRY  (X5) = O.SEQ                                           GRA        2342     A
GRA      *                                                                       GRA        2343     A
GRA      *         EXIT   N.MR, N.ST UPDATED                                     GRA        2344     A
GRA                                                                              GRA        2345     A
GRA       CMRA     BFMW   D,(BM,LD,ST)                                           GRA        2346     A
GRA                                                                              GRA        2347     A
GRA       CMR0     SX6    B4          N.LD = SLD                                 GRA        2348     A
GRA                SX7    B7                                                     GRA        2349     A
GRA                SA6    A1                                                     GRA        2350     A
GRA                SA7    A1+B1                                                  GRA        2351     A
GRA                                                                              GRA        2352     A
GRA       CMR      ROUTINE                                                       GRA        2353     A
GRA                SA1    N.LD                                                   GRA        2354     A
GRA                SA2    N.ST                                                   GRA        2355     A
GRA                SA3    O.RCT                                                  GRA        2356     A
GRA                SA4    X5+2        DI = O.SEQ + 2                             GRA        2357     A
GRA                SB3    B0          NMR = 0       */ MEM REFS IN BLOCK         GRA        2358     A
GRA                SA5    CMRA                                                   GRA        2359     A
GRA                SB4    X1          SLD = N.LD    */ SUM OF *LD*"S             GRA        2360     A
GRA                BX0    X5                                                     GRA        2361     A
GRA                SB5    B0          NST = 0       */ *ST*"S IN BLOCK           GRA        2362     A
GRA                SB6    X3                                                     GRA        2363     A
GRA                SB7    X2          MST = N.ST    */ MAX *ST*"S SO FAR         GRA        2364     A
GRA                EQ     CMR4                                                   GRA        2365     A
GRA                                                                              GRA        2366     A
GRA      *         PROCESS BOUNDARY MARKER                                       GRA        2367     A
GRA                                                                              GRA        2368     A
GRA       CMR1     SB3    B3-B5       NMR = NMR - NST                            GRA        2369     A
GRA                LE     B5,B7,CMR2  N.ST = MAX( NST , N.ST )                   GRA        2370     A
GRA                SB7    B5                                                     GRA        2371     A
GRA       CMR2     SA5    A4-2        R1 = DI - 2                                GRA        2372     A
GRA                UX6    B2,X5                                                  GRA        2373     A
GRA                SB4    B4+B3       SLD = SLD + NMR                            GRA        2374     A
GRA                ZR     B2,CMR0     IF OC[R1] = OC.EOQ                         GRA        2375     A
GRA                                                                              GRA        2376     A
GRA                SB3    B0          NMR = 0                                    GRA        2377     A
GRA                SB5    B0          NST = 0                                    GRA        2378     A
GRA                                                                              GRA        2379     A
GRA      *         ADVANCE TO NEXT INSTRUCTION, CHECK FOR MEMORY REFERENCE       GRA        2380     A
GRA                                                                              GRA        2381     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA       CMR4     SA4    A4+4        DI = DI + 4                                    GRA        2382     A
GRA                BX7    X0*X4                                                      GRA        2383     A
GRA                LX4    59-D.BMP                                                   GRA        2384     A
GRA                ZR     X7,CMR4     IF ^(BM,LD,ST)[DI]                             GRA        2385     A
GRA                                                                                  GRA        2386     A
GRA                MI     X4,CMR1     IF BM[DI]                                      GRA        2387     A
GRA                LX4    D.BMP-D.L2P                                                GRA        2388     A
GRA                MI     X4,CMR4     IF L2[DI]     */ LEVEL 2 LD/ST                 GRA        2389     A
GRA                                                                                  GRA        2390     A
GRA      *         MEMORY REF, COUNT IT IF NOT ASSIGNED TO A REGISTER               GRA        2391     A
GRA                                                                                  GRA        2392     A
GRA                SA3    A4+B1       LI = DI + 1;  I = CO[LI]                       GRA        2393     A
GRA                LX7    59-D.LDP                                                   GRA        2394     A
GRA                SA2    B6+X3       CI = RCT(I)                                    GRA        2395     A
GRA                LX2    59-CA.RAP                                                  GRA        2396     A
GRA                PL     X2,CMR6     IF ^RA[CI]    */ NOT IN A REG                  GRA        2397     A
GRA                                                                                  GRA        2398     A
GRA      *         COUNT STORE IF WE CANT MOVE IT OUT                               GRA        2399     A
GRA                                                                                  GRA        2400     A
GRA                MI     X7,CMR4     IF LD[DI]                                      GRA        2401     A
GRA                LX2    CA.RAP-CA.LXP                                              GRA        2402     A
GRA                PL     X2,CMR4     IF ^LX[CI]    */ DEAD ON EXIT                  GRA        2403     A
GRA                LX2    CA.LXP-CA.PSPP                                             GRA        2404     A
GRA                MI     X2,CMR4     IF PSP[DI]    */ POST STORE POSSIBLE           GRA        2405     A
GRA                                                                                  GRA        2406     A
GRA       CMR6     SB3    B3+B1       NMR = NMR + 1                                  GRA        2407     A
GRA                MI     X7,CMR4     IF LD[DI]                                      GRA        2408     A
GRA                SB5    B5+B1       NST = NST + 1                                  GRA        2409     A
GRA                EQ     CMR4                                                       GRA        2410     A
GRA       DAA      TITLE  DAA - DETERMINE A-ASSIGNMENTS ( PREFETCHS )               GRA        2411     A
GRA      **        DAA - DETERMINE A-ASSIGNMENTS ( PREFETCHS )                      GRA        2412     A
GRA      *                                                                           GRA        2413     A
GRA      *         THE CRITERIA FOR PREFETCHING FALL INTO TWO CATEGORIES -          GRA        2414     A
GRA      *         1) THAT THE LOOP BE SUFFICIENTLY SMALL SO WE CAN DO IT.          GRA        2415     A
GRA      *         2) THE RESTRICTIONS THAT THE PREFETCH CANIDATES MUST MEET.       GRA        2416     A
GRA      *         SOME OF THE RESTRICTIONS ARE NATURAL, WHILE OTHERS ARE           GRA        2417     A
GRA      *         ARTIFICAL AND COULD BE RELAXED IF WE HAD MORE INFORMATION.       GRA        2418     A
GRA      *                                                                           GRA        2419     A
GRA      *         EXIT   (NAA) = N. A-ASSIGNMENTS                                  GRA        2420     A
GRA      *                ACT(I) = 24/RCT-INC INFO,18/,18/X-REGNO OF PREFETCH      GRA        2421     A
GRA                                                                                  GRA        2422     A
GRA       DAAA     BFMW   D,(LD,BM)                                                 GRA        2423     A
GRA                BFMW   CA,(RA,CON)                                               GRA        2424     A
GRA       DAAB     BFMW   CA,(RA,VC)                                                GRA        2425     A
GRA                                                                                  GRA        2426     A
GRA       DAA      ROUTINE                                                          GRA        2427     A
GRA                SA1    MLW                                                       GRA        2428     A
GRA                SA2    N.EBB                                                     GRA        2429     A
GRA                SX6    X1-8                                                      GRA        2430          I
          -FCC0603
GRA                PL     X6,DAA      IF MLW > 7    */ LOOP IS TOO WIDE              GRA        2431          I
          -FCC0603

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SX5    X1-2                                                  GRA         2432          I
          -FCC0603
GRA                SX6    X1-8                                                  FCC0603       23     A
GRA                MI     X6,DAA0     IF MLW < 8                                FCC0603       24     A
GRA                NZ     X6,DAA      IF MLW > 8                                FCC0603       25     A
GRA                SB2    X2                                                    FCC0603       26     A
GRA                GT     B2,B1,DAA   IF N.EBB > 1                              FCC0603       27     A
GRA                SA3    L.SEQ                                                 FCC0603       28     A
GRA                SX6    X3-240B                                               FCC0603       29     A
GRA                PL     X6,DAA      IF L.SEQ > 240B  */ MAGIC NUMBER          FCC0603       30     A
GRA       DAA0     SX5    X1-2                                                  FCC0603       31          I
          -FCC1127
GRA                MI     X5,DAA      IF MLW < 2    */ MOVE LOOP                GRA         2433          I
          -FCC1127
GRA                SX7    X2-3                                                  GRA         2434          I
          -FCC1127
GRA       DAA0     SX7    X2-3                                                  FCC1127        4     A
GRA                PL     X7,DAA      IF N.EBB > 2  */ > 1 EBB + 1 LATCH NODE   GRA         2435     A
GRA                                                                             GRA         2436     A
GRA                SA3    N.LD                                                  GRA         2437     A
GRA                SA4    N.ST                                                  GRA         2438     A
GRA                SA5    N.LJ                                                  GRA         2439     A
GRA                SX6    X3-8                                                  GRA         2440     A
GRA                SX7    X4-3                                                  GRA         2441     A
GRA                PL     X6,DAA      IF N.LD > 7                               GRA         2442     A
GRA                PL     X7,DAA      IF N.ST > 2                               GRA         2443     A
GRA                SX6    B1                                                    GRA         2444     A
GRA                IX5    X5-X6                                                 GRA         2445     A
GRA                NZ     X5,DAA      IF N.LJ " 1   */ ONLY *LBJ* IS A *JPX*    GRA         2446     A
GRA                SX7    X1-2                                                  FCC1127        5     A
GRA                PL     X7,DAA0A    IF MLW \ 2                                FCC1127        6     A
GRA                IX6    X3-X4                                                 FCC1127        7     A
GRA                ZR     X6,DAA      IF N.LD = N.ST  */ A MOVE LOOP            FCC1127        8     A
GRA                                                                             GRA         2447     A
GRA       *        DETERMINE MAA, THE MAX NUMBER OF PREFETCHS WE WILL ASSIGN.   GRA         2448     A
GRA                                                                             GRA         2449     A
GRA                SA1    MRA                                                   GRA         2450          I
          -FCC1127
GRA       DAA0A    SA1    MRA                                                   FCC1127        9     A
GRA                SX2    76B                                                   GRA         2451     A
GRA                LX1    -16                                                   GRA         2452     A
GRA                BX6    X2*X1       ALR = SHIFT(MRA,-16) & 76B  */ AVAIL LDREGS  GRA       2453     A
GRA                CX5    X6          NLR = COUNT( ALR )                        GRA         2454     A
GRA                IX6    X5-X3                                                 GRA         2455     A
GRA                LX5    1                                                     GRA         2456     A
GRA                IX7    X5-X3       MAA = 2*NLR - N.LD  */ MAX A-ASSGNMNTS    GRA         2457     A
GRA                                                                             GRA         2458     A
GRA                ZR     X7,DAA      IF MAA = 0                                GRA         2459          I
          -FCC0603
GRA                MI     X7,DAA      IF MAA < 0                                GRA         2460          I
          -FCC0603
GRA                SX1    X7-5                                                  GRA         2461          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

          -FCC0603
GRA       +         MI     X1,*+1      MAA = MIN( 4 , MAA )                          GRA         2462          I
          -FCC0603
GRA                 SX7    4                                                         GRA         2463          I
          -FCC0603
GRA                 SA1    MARA                                                      FCC0603       32     A
GRA                 IX2    X1-X7                                                     FCC0603       33     A
GRA                 AX2    59                                                        FCC0603       34     A
GRA                 BX1    X2*X1                                                     FCC0603       35     A
GRA                 BX2    -X2*X7                                                    FCC0603       36     A
GRA                 IX7    X1+X2       MAA = MIN( MARA , MAA )                       FCC0603       37     A
GRA                 SB2    X7                                                        FCC0603       38     A
GRA                 LE     B2,DAA      IF MAA @ 0                                    FCC0603       39     A
GRA                 PL     X6,DAA1     IF NLR \ N.LD                                 GRA         2464     A
GRA                 BX6    X7                                                        GRA         2465     A
GRA                 AX6    2                                                         GRA         2466     A
GRA                 ZR     X6,DAA1     MAA = MIN( 3 , MAA )                          GRA         2467     A
GRA                 SX7    3                                                         GRA         2468     A
GRA        DAA1     SA7    MAA                                                       GRA         2469     A
GRA                 SX6    2                                                         GRA         2470     A
GRA                 SA6    BSI         BSI = 2                                       GRA         2471     A
GRA                 CALL   RTB         GET POINTERS TO HEADER NODE                   GRA         2472          I
          -CCGNC
GRA                 CALL   RTB#        GET POINTERS TO HEADER NODE                   CCGNC        651     A
GRA                                                                                  CCG25          6     A
GRA                 CALL   AAV         ASSIGN ARRAY VARIABLES TO A/X REGISTERS       CCG25          7     A
GRA                 SA5    MAA                                              002320   CCG41          2     A
GRA                 SB5    X5                                               002330   CCG41          3     A
GRA                 LE     B5,DAA      IF MAA LE 0,  RETURN                 002340   CCG41          4     A
GRA        DAA      SPACE  2,10                                                      GRA         2473     A
GRA       *         SELECT CANIDATES FROM THE INDEXED LD"S IN THE HEADER NODE.       GRA         2474     A
GRA       *         FORM *ACT* - 24/RCT-INC INFO,18/INC FIRST BIAS,18/RCT-INDEX      GRA         2475     A
GRA                                                                                  GRA         2476     A
GRA                 SA2    O.UDT                                                     GRA         2477     A
GRA                 SA3    O.RCT                                                     GRA         2478     A
GRA                 SA5    O.SEQ                                                     GRA         2479     A
GRA                 SA1    DAAA                                                      GRA         2480     A
GRA                 SA4    A1+B1                                                     GRA         2481     A
GRA                 SB5    B0          J = 0         */ ACT STORE INDEX              GRA         2482     A
GRA                 SB6    X3                                                        GRA         2483     A
GRA                 BX0    X4                                                        GRA         2484     A
GRA                 SB7    X2                                                        GRA         2485     A
GRA                 SA4    X5+6        DI = O.SEQ + 4 + 2  */ SKIP PAST *LAB*        GRA         2486     A
GRA                 SB4    X5+2                     (B4) = TXT+2                     GRA         2487     A
GRA                                                                                  GRA         2488     A
GRA        DAA2     SA4    A4+4        DI = DI + 4                                   GRA         2489     A
GRA                 BX6    X1*X4                                                     GRA         2490     A
GRA                 LX4    59-D.BMP                                                  GRA         2491     A
GRA                 ZR     X6,DAA2     IF ^( LD[DI] ! BM[DI] )                       GRA         2492     A
GRA                                                                                  GRA         2493     A
GRA                 MI     X4,DAA4     IF BM[DI]                                     GRA         2494     A
GRA                 LX4    D.BMP-D.L2P                                               GRA         2495     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                MI     X4,DAA2     IF L2[DI]     */ LEVEL 2 LD                    GRA         2496     A
GRA                                                                                  GRA         2497     A
GRA                SA5    A4+B1       LI = DI + 1;  I = CO[LI]                       GRA         2498     A
GRA                SA3    B6+X5       CI = RCT(I)                                    GRA         2499     A
GRA                SA2    B7+X3       UI = UDT( UDI[CI] )                            GRA         2500     A
GRA                LX2    59-UD.DEFRP                                                GRA         2501     A
GRA                MI     X2,DAA2     IF DEFR[UI]   */ STORE INTO SAME CLASS         GRA         2502     A
GRA                LX3    59-CA.ACP                                                  GRA         2503     A
GRA                PL     X3,DAA2     IF ^AC[CI]    */ NOT AN ADDRESS CANIDATE       GRA         2504     A
GRA                LX3    CA.ACP-CA.NOCCP                                            FCC0448      103     A
GRA                MX6    -CA.NOCCL+1                                                FCC0448      104     A
GRA                BX3    -X6*X3                                                     FCC0448      105     A
GRA                NZ     X3,DAA2     IF NOCC[CI] > 1  */ POSSIBLE A(I);I=I+1;A(I    FCC0448      106     A
GRA                                                                                  FCC0448      107     A
GRA                SA2    A3+B1       C2W = RCT(I+1)                                 GRA         2505     A
GRA                LX2    -IH.RFP     R = RF[C2W]                                    GRA         2506     A
GRA                                                                                  GRA         2507     A
GRA      *         LOOK AT INDEX OF *LD* FOR INFO ABOUT THE INCREMENT VALUE.         GRA         2508     A
GRA                                                                                  GRA         2509     A
GRA                SB2    X2+2                                                       GRA         2510     A
GRA                SA3    B6+B2       CCW = RCT(R+2)                                 GRA         2511     A
GRA                LX3    59-CC.I1P                                                  GRA         2512     A
GRA                PL     X3,DAA2     IF ^I1[CCW]   */ NOT *RD*                      GRA         2513     A
GRA                LX3    CC.I1P-CC.I2P                                              GRA         2514     A
GRA                MI     X3,DAA2     IF I2[CCW]    */ MORE THAN 1 INC OF INDEX      GRA         2515     A
GRA                LX3    1+CC.I2P-CC.INCP  IP = INC[CCW] */ RCT INDEX OF INC VAL    GRA         2516     A
GRA                SA2    B6+X3       CJ = RCT(IP)                                   GRA         2517     A
GRA                BX6    X0*X2                                                      GRA         2518     A
GRA                ZR     X6,DAA2     IF ^( RA[CJ] ! CON[CJ] )                       GRA         2519     A
GRA                                                                                  GRA         2520     A
GRA                LX3    CC.INCP                                                    GRA         2521     A
GRA                MX7    -CC.INCP                                                   GRA         2522     A
GRA                BX3    X7*X3                                                      GRA         2523          I
          -FCC1127
GRA                BX6    X3+X5                                                      GRA         2524          I
          -FCC1127
GRA                BX7    X7*X3                                                      FCC1127       10     A
GRA                BX6    X7+X5                                                      FCC1127       11     A
GRA                SA5    A4-B1       R2 = DI - 1                                    GRA         2525     A
GRA                LX5    -IH.RFP                                                    GRA         2526     A
GRA                SA3    B4+X5       DIP = TXT(RF[R2]+2)                            GRA         2527          I
          -FCC1127
GRA                LX3    58-D.TYP                                                   GRA         2528          I
          -FCC1127
GRA                PL     X3,DAA2     IF TYPE[DIP] " III  */ IA/IS INC FIRST         GRA         2529          I
          -FCC1127
GRA                LX3    1+D.TYP-D.LDP                                              GRA         2530          I
          -FCC1127
GRA                MI     X3,DAA3     IF LD[DIP]    */ NOT INCREMENT FIRST           GRA         2531          I
          -FCC1127
GRA                MX7    -IH.CAL                                                    GRA         2532          I
          -FCC1127

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                 SA3    A3-B1       R2P = DIP - 1                                  GRA          2533          I
          -FCC1127
GRA                 LX7    IH.CAP                                                     GRA          2534          I
          -FCC1127
GRA                 BX5    -X7*X3                                                     GRA          2535          I
          -FCC1127
GRA                 BX6    X5+X6       BIAS = CA[R2P]   */ INC FIRST BIAS             GRA          2536          I
          -FCC1127
GRA                 SA5    B4+X5       DIP = TXT(RF[R2]+2)                            FCC1127        12     A
GRA                 LX5    59-D.LDP                                                   FCC1127        13     A
GRA                 MI     X5,DAA3     IF LD[DIP]    */ ^ INC FIRST                   FCC1127        14     A
GRA                 LX2    59-CA.CONP                                                 FCC1127        15     A
GRA                 PL     X2,DAA2     IF ^CON[CJ]   */ INC VALUE ^ A CON             FCC1127        16     A
GRA                                                                                   FCC1127        17     A
GRA      *          INC-FIRST-BIAS = CA[C2J] * IM[CCW]                                FCC1127        18     A
GRA                                                                                   FCC1127        19     A
GRA                 SA5    A2+B1       C2J = RCT(IP+1)                                FCC1127        20     A
GRA                 LX3    59-CC.IMP                                                  FCC1127        21     A
GRA                 MX7    -IH.CAL                                                    FCC1127        22     A
GRA                 AX3    59                                                         FCC1127        23     A
GRA                 LX7    IH.CAP                                                     FCC1127        24     A
GRA                 BX3    X3-X5                                                      FCC1127        25     A
GRA                 BX5    -X7*X3                                                     FCC1127        26     A
GRA                 BX6    X5+X6       BIAS = CA[C2J] * IM[CCW]                       FCC1127        27     A
GRA                                                                                   GRA          2537     A
GRA       DAA3      SA6    ACT+B5      ACT(J) = ACW(INCI[CCW],BIAS,I)                 GRA          2538     A
GRA                 SB5    B5+B1       J = J + 1                                      GRA          2539     A
GRA                 SX7    B5-7                                                       GRA          2540     A
GRA                 MI     X7,DAA2     IF J < 7                                       GRA          2541     A
GRA       DAA       SPACE  2,10                                                       GRA          2542     A
GRA      *          AT THIS POINT WE SHOULD RUN A CRITICAL PATH CALCULATION AND       GRA          2543     A
GRA      *          SORT THE CANIDATE TABLE. FOR THE PRESENT WE OMIT IT.              GRA          2544     A
GRA                                                                                   GRA          2545     A
GRA       DAA4      ZR     B5,DAA      IF J = 0      */ NO CANIDATES                  GRA          2546     A
GRA                 SA5    O.SYM                                                      GRA          2547          I
          -CCG1
GRA                 SA5    =XO.SYM                                                    CCG1           96     A
GRA                 SA4    MAA                                                        GRA          2548     A
GRA                 SA0    X5-1                                                       GRA          2549          I
          -CCG1
GRA                 SA0    X5+1                                                       CCG1           97     A
GRA                 SB4    B0          FI = 0                                         GRA          2550     A
GRA                 MX7    0                                                          GRA          2551     A
GRA                 SA7    ACT+B5      ACT(J) = 0                                     GRA          2552     A
GRA                 SB7    X4                                                         GRA          2553     A
GRA                 SX0    B1                                                         GRA          2554     A
GRA                 SB5    B0          SI = 0                                         GRA          2555     A
GRA                 LX0    CA.NOCCP                                                   GRA          2556     A
GRA                                                                                   GRA          2557     A
GRA      *          PRUNE THE CANIDATE TABLE TO REMOVE THOSE WITH VARIABLE            GRA          2558     A
GRA      *          OR LARGE CONSTANT INCREMENTS ( AVOID MODE 1 AT EXECUTION )        GRA          2559     A
GRA                                                                                   GRA          2560     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA        DAA5     SA1    =XOPTLVL                                                GRA         2561          I
          -FEAT121A
GRA                 SX7    X1-3                                                    GRA         2562          I
          -FEAT121A
GRA        DAA5     SA1    =XUOFLAG                                                FEAT121A     125          I
          -CCG1
GRA        DAA5     SA1    =XCO.UO                                                 CCG1          98          I
          -CCGNC
GRA        DAA5     SA1    =XHO$UO                                                 CCGNC        652     A
GRA                 SA5    ACT+B4                                                  GRA         2563     A
GRA                 ZR     X5,DAA9                                                 GRA         2564     A
GRA                 SB2    X5+B1        I = CO[ACT(FI)]                            GRA         2565     A
GRA                 LX5    -CC.INCP     IP = INC[ACT(FI)]                          GRA         2566     A
GRA                 PL     X7,DAA7      IF OPTLVL > 2 */ TAKE ALL IF OPT > 2       GRA         2567          I
          -FEAT121A
GRA                 NZ     X1,DAA7      IF UOFLAG " 0 */ TAKE ALL IF *UO* SELECTED FEAT121A     126          I
          -CCG1
GRA                 NZ     X1,DAA7      IF CO.UO  " 0 */ TAKE ALL IF *UO* SELECTED CCG1          99          I
          -CCGNC
GRA                 NZ     X1,DAA7      IF HO$UO  " 0 */ TAKE ALL IF *UO* SELECTED CCGNC        653     A
GRA                                                                                GRA         2568     A
GRA                 SA4    B6+X5        CI = RCT(IP)                               GRA         2569     A
GRA                 LX4    59-CA.CONP                                              GRA         2570     A
GRA                 PL     X4,DAA8      IF ^CON[CI]   */ REJECT IF VARIABLE INC    GRA         2571     A
GRA                 SA4    A4+B1        C2W = RCT(IP+1)                            GRA         2572     A
GRA                 LX4    -IH.CAP      C = CA[C2W]   */ CONSTANT INC VALUE        GRA         2573     A
GRA                 SX7    X4-3                                                    GRA         2574     A
GRA                 MI     X7,DAA7      IF C @ 2                                   GRA         2575     A
GRA                 SX6    X4-MAX.INC                                              GRA         2576     A
GRA                 PL     X6,DAA8      IF C > MAX.INC  */ CON INC TOO BIG         GRA         2577     A
GRA                                                                                GRA         2578     A
GRA      *          CHECK LOCATION OF SYMBOL, IF IN LABELED COMMON OR LOCAL,       GRA         2579     A
GRA      *          ASSUME THAT THEIR WILL BE AT LEAST MAX.INC STORAGE ABOVE IT.   GRA         2580     A
GRA                                                                                GRA         2581     A
GRA                 SA4    B6+B2        C2W = RCT(I+1)                             GRA         2582     A
GRA                 LX4    -IH.IHP+1    H = IH[C2W]                                GRA         2583          I
          -CCG1
GRA                 SB2    X4                                                      GRA         2584          I
          -CCG1
GRA                 SA3    A0-B2        WORDB = [O.SYM-1-2*H]                      GRA         2585          I
          -CCG1
GRA                 LX3    59-P.FPB                                                GRA         2586          I
          -CCG1
GRA                 MI     X3,DAA8      IF FP[WORDB]  */ SYM IS AN F.P.            GRA         2587          I
          -CCG1
GRA                 LX3    1+P.FPB-P.RL                                            GRA         2588          I
          -CCG1
GRA                 MX7    -L.RL                                                   GRA         2589          I
          -CCG1
GRA                 BX6    -X7*X3                                                  GRA         2590          I
          -CCG1
GRA                 SX7    X6-2                                                    GRA         2591          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
          -CCG1
GRA                NZ     X7,DAA7      IF RT " 2     */ NOT IN COMMON                    GRA        2592          I
          -CCG1
GRA                LX3    P.RL-P.RB                                                      GRA        2593          I
          -CCG1
GRA                MX7    -L.RB                                                          GRA        2594          I
          -CCG1
GRA                BX6    -X7*X3       RBN = RB[WORDB]  */ RELOC BLOCK NUMBER            GRA        2595          I
          -CCG1
GRA                SA2    =XORGTAB-1+X6  CBN = ORGTAB(RBN)  */ BLOCK NAME AND LEN        GRA        2596          I
          -CCG1
GRA                AX2    54                                                             GRA        2597          I
          -CCG1
GRA                SX7    X2+77B-1R                                                      GRA        2598          I
          -CCG1
GRA                ZR     X7,DAA8      IF CBN = * // * COMMON                            GRA        2599          I
          -CCG1
GRA                SB2    X4           H = IH[C2W]                                       CCG1        100     A
GRA                LX4    1                                                              CCG1        101     A
GRA                SB2    B2+X4                                                          CCG1        102     A
GRA                SA3    A0+B2        WORDB = SYM(3*IH+1)                               CCG1        103     A
GRA                LX3    59-WB.FPP                                                      CCG1        104     A
GRA                MI     X3,DAA8      IF FP[WORDB]  */ LOCATION UNKNOWN                 CCG1        105     A
GRA                LX3    WB.FPP-WB.COMP                                                 CCG1        106     A
GRA                PL     X3,DAA7      IF ^COM[WORDB]  */ NOT IN COMMON                  CCG1        107     A
GRA                LX3    1+WB.COMP-WB.RBP                                               CCG1        108          I
          -CCG16
GRA                MX7    -WB.RBL                                                        CCG1        109          I
          -CCG16
GRA                BX6    -X7*X3       RBN = RB[WORDB]                                   CCG1        110          I
          -CCG16
GRA                SA3    A3+B1        WORDC = WORDB + 1                                 CCG16       215     A
GRA                MX7    -WC.RBL                                                        CCG16       216     A
GRA                LX3    -WC.RBP                                                        CCG16       217     A
GRA                BX6    -X7*X3       RBN = RB[WORDC]                                   CCG16       218     A
GRA       .FTN     IFEQ   HC.ID,2,2                                           001210     FCC2135       1     A
GRA                SA1    =XBLNKCOM                                          001220     FCC2135       2     A
GRA                BX6    X6-X1                                              001230     FCC2135       3     A
GRA                ZR     X6,DAA8      IF RBN = 0    */ IN // COMMON                     CCG1        111     A
GRA                                                                                      GRA        2600     A
GRA       *        ADD NAME TO FINAL CANIDATE LIST                                       GRA        2601     A
GRA                                                                                      GRA        2602     A
GRA       DAA7     SA4    B6+X5        CIP = RCT(IP)                                     GRA        2603     A
GRA                LX5    CC.INCP                                                        GRA        2604     A
GRA                IX7    X0+X4        NOCC[CIP] = NOCC[CIP] + 1                         GRA        2605     A
GRA                BX6    X5                                                             GRA        2606     A
GRA                SA6    ACT+B5       ACT(SI) = ACT(FI)                                 GRA        2607     A
GRA                SA7    A4                                                             GRA        2608     A
GRA                SB5    B5+B1        SI = SI + 1                                       GRA        2609     A
GRA                EQ     B5,B7,DAA9   IF SI = MAA                                       GRA        2610     A
GRA                                                                                      GRA        2611     A
GRA       DAA8     SB4    B4+B1        FI = FI + 1                                       GRA        2612     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                EQ     DAA5                                                    GRA        2613     A
GRA                                                                               GRA        2614     A
GRA       DAA9     ZR     B5,DAA      IF SI = 0                                   GRA        2615     A
GRA                                                                               GRA        2616     A
GRA                SA1    MRA                                                     GRA        2617     A
GRA                SB3    20B         REGN = 20B    */ X0, X1,...                 GRA        2618     A
GRA                SB4    B0          J = 0                                       GRA        2619     A
GRA                SX0    B1                                                      GRA        2620     A
GRA                                                                               GRA        2621     A
GRA      *         NOW MARK THE PREFETCH ASSIGNMENTS IN *RCT* AND *RAT*           GRA        2622     A
GRA                                                                               GRA        2623     A
GRA       DAA10    SA2    ACT+B4      I = CO[ACT(J)]                              GRA        2624     A
GRA                SA4    DAAB                                                    GRA        2625     A
GRA                SA3    X2+B6       CI = RCT(I)                                 GRA        2626     A
GRA                BX6    X4+X3       (RA,VC)[CI] = 1                             GRA        2627     A
GRA                SA6    A3                                                      GRA        2628     A
GRA                SA4    A3+B1       C2W = RCT(I+1)                              GRA        2629     A
GRA                LX2    -IH.CAP                                                 GRA        2630     A
GRA                SB2    X2                                                      GRA        2631     A
GRA                LX4    -IH.CAP                                                 GRA        2632     A
GRA                SX5    B2+X4                                                   GRA        2633     A
GRA                MX7    -IH.CAL                                                 GRA        2634     A
GRA                BX5    -X7*X5      CA[C2W] = CA[C2W] + CA[ACT(J)]              GRA        2635     A
GRA                LX2    IH.CAP                                                  GRA        2636     A
GRA                BX7    X7*X4                                                   GRA        2637     A
GRA                BX6    X7+X5                                                   GRA        2638     A
GRA                LX6    IH.CAP                                                  GRA        2639     A
GRA                SA6    A4                                                      GRA        2640     A
GRA                LX4    IH.CAP-IH.RFP     R = RF[C2W]                           GRA        2641     A
GRA                SA5    B6+X4       CJ = RCT(R)                                 GRA        2642     A
GRA                MX7    -CA.SUSEL                                               GRA        2643     A
GRA                LX5    -CA.SUSEP                                               GRA        2644     A
GRA                IX6    X5-X0       SUSE[CJ] = SUSE[CJ] - 1                     GRA        2645     A
GRA                BX7    -X7*X6                                                  GRA        2646     A
GRA                LX6    CA.SUSEP                                                GRA        2647     A
GRA                SA6    A5                                                      GRA        2648     A
GRA                SX7    X7-1                                                    GRA        2649     A
GRA                NZ     X7,DAA11    IF SUSE[CJ] " 1                             GRA        2650     A
GRA                                                                               GRA        2651     A
GRA      *         USES OF INCREMENT VARIABLE = 1 , DEALLOCATE THE B-REGISTER     GRA        2652     A
GRA      *         AND MARK THE VAR AS DEAD.                                      GRA        2653     A
GRA                                                                               GRA        2654     A
GRA                SX7    X6                                                      GRA        2655     A
GRA                NZ     X7,DAA10A   IF UDI[CJ] " 0  */ PROGRAMMER VAR           GRA        2656          I
          -FCC0414
GRA                LX6    59-CA.CONP                                              GRA        2657          I
          -FCC0414
GRA                LX6    59-CA.LXP                                               FCC0414       6     A
GRA                MI     X6,DAA11    IF LX[CJ]     */ LIVE ON EXIT FROM THE LOOP FCC0414       7     A
GRA                NZ     X7,DAA10A   IF UDI[CJ] " 0  */ PROGRAMMER VARIABLE      FCC0414       8     A
GRA                LX6    CA.LXP-CA.CONP                                          FCC0414       9     A
GRA                MI     X6,DAA10A   IF CON[CJ]                                  GRA        2658     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                                                                                GRA      2659   A
GRA                SA4    A5+B1                                                    GRA      2660   A
GRA                MX7    -T.REGL                                                  GRA      2661   A
GRA                LX4    -IH.CAP                                                  GRA      2662   A
GRA                SB2    X4          C = CA[RCT(R+1)]                             GRA      2663   A
GRA                SA4    O.TET                                                    GRA      2664   A
GRA                LX7    T.REGP                                                   GRA      2665   A
GRA                SA4    X4+B2                                                    GRA      2666   A
GRA                BX6    X7*X4       REG[TET(C)] = 0                              GRA      2667   A
GRA                SA6    A4                                                       GRA      2668   A
GRA                                                                                GRA      2669   A
GRA       DAA10A   SA4    A5+2        CCW = RCT(R+2)                               GRA      2670   A
GRA                MX7    -CC.REG1L                                                GRA      2671   A
GRA                SX6    B0                                                       GRA      2672   A
GRA                LX4    -CC.REG1P                                                GRA      2673   A
GRA                BX4    -X7*X4      REG = REG1[CCW]                              GRA      2674   A
GRA                SB2    X4                                                       GRA      2675   A
GRA                SA6    RAT+X4      RAT(REG) = 0                                 GRA      2676   A
GRA                LX7    B2,X0                                                    GRA      2677   A
GRA                BX1    X7+X1       MRA = MRA ! SHIFT(1,REG)                     GRA      2678   A
GRA                SA4    NAB                                                      GRA      2679   A
GRA                SX7    X4+B1       NAB = NAB + 1                                GRA      2680   A
GRA                SA7    A4                                                       GRA      2681   A
GRA                SA4    SEEA                                                     GRA      2682   A
GRA                LX5    CA.SUSEP                                                 GRA      2683   A
GRA                BX6    X4+X5       KD[CJ] = 1    */ SET TO KILL THIS DEF        GRA      2684   A
GRA                SA6    A5                                                       GRA      2685   A
GRA                                                                                GRA      2686   A
GRA      *         SELECT A LD-REG AND MARK THE ASSIGNMENT IN *RCT* AND *RAT*      GRA      2687   A
GRA                                                                                GRA      2688   A
GRA       DAA11    SB3    B3+1        REGN = REGN + 1                              GRA      2689   A
GRA                LX7    B3,X0       BIT = SHIFT(1,REGN)                          GRA      2690   A
GRA                BX6    X7*X1                                                    GRA      2691   A
GRA                ZR     X6,DAA11    IF MRA & BIT = 0  */ LD-REG NOT AVAIL        GRA      2692   A
GRA                                                                                GRA      2693   A
GRA                SA4    A3+2        CCW = RCT(I+2)                               GRA      2694   A
GRA                BX1    -X7*X1      MRA = MRA & ^BIT                             GRA      2695   A
GRA                SX6    X2                                                       GRA      2696   A
GRA                SA6    RAT+B3      RAT(REGNO) = I                               GRA      2697   A
GRA                SX3    B3                                                       GRA      2698   A
GRA                BX2    X2-X6                                                    GRA      2699   A
GRA                BX7    X2+X3       CO[ACT(J)] = REGNO  */ SAVE X-REGNO IN ACT   GRA      2700   A
GRA                SA7    A2                                                       GRA      2701   A
GRA                LX3    CC.REG1P                                                 GRA      2702   A
GRA                BX7    X3+X4       REG1[CCW] = REGN                             GRA      2703   A
GRA                SA7    A4                                                       GRA      2704   A
GRA                SB4    B4+B1       J = J + 1                                    GRA      2705   A
GRA                LT     B4,B5,DAA10 IF J < SI                                    GRA      2706   A
GRA                                                                                GRA      2707   A
GRA                SX6    B5                                                       GRA      2708   A
GRA                BX7    X1                                                       GRA      2709   A
GRA                SA6    NAA         NAA = SI                                     GRA      2710   A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SA7    MRA                                                  GRA        2711    A
GRA                EQ     DAA                                                  GRA        2712    A
GRA       AAV      TITLE  AAV - ASSIGN ARRAY VARIABLES TO REGISTERS            CCG25         8    A
GRA      **        AAV - ASSIGN ARRAY VARIABLES TO X-REGISTERS                 CCG25         9    A
GRA      *                                                                     CCG25        10    A
GRA      *         CHECK LOOP FOR STATEMENTS OF THE FORM A(I) = A(I) OP EXPR   CCG25        11    A
GRA      *         WHERE I IS INVARIANT AND NO OTHER REFS TO A.  IF SO, THEN   CCG25        12    A
GRA      *         ASSIGN A(I) TO AN X-REGISTER WITH THE ADDRESS IN THE        CCG25        13    A
GRA      *         CORRESPONDING A-REGISTER.                                   CCG25        14    A
GRA                                                                            CCG25        15    A
GRA       CA.NO=2P EQU    CA.NOCCP+1                                           CCG25        16    A
GRA       CA.NO=2L EQU    1                                                    CCG25        17    A
GRA       AAVA     BFMW   CA,(AC,NO=2,DEF,USE)                                 CCG25        18    A
GRA                                                                            CCG25        19    A
GRA       AXCT     BSSZ   4           A/X CANIDATE TABLE   24/0,18/RCT-ORD,18/REG  CCG25    20    A
GRA                                                                            CCG25        21    A
GRA       AAV      ROUTINE                                                     CCG25        22    A
GRA                SA1    HO$OPT                                               CCG0093       5    A
GRA                PL     X1,AAV      IF OPT NE 2                              CCG0093       6    A
GRA                SA1    N.ENL                                                CCG25        23    A
GRA                SB5    B0          J = 0                                    CCG25        24    A
GRA                AX1    1                                                    CCG25        25    A
GRA                NZ     X1,AAV5     IF N.ENL > 1  */ MORE THAN 1 EXIT NODE   CCG25        26    A
GRA                                                                            CCG25        27    A
GRA      *         SCAN *RCT* FOR CANIDATES                                    CCG25        28    A
GRA                                                                            CCG25        29    A
GRA                SA4    O.RCT                                                CCG25        30    A
GRA                SA5    LCT                                                  CCG25        31    A
GRA                SA1    AAVA                                                 CCG25        32    A
GRA                SB6    X4                                                   CCG25        33    A
GRA                SB4    X5+B1                                                CCG25        34    A
GRA                BX0    X1                                                   CCG25        35    A
GRA                SA5    B6+X5       CI = RCT + LCT                           CCG25        36    A
GRA                                                                            CCG25        37    A
GRA       AAV2     SA5    A5-3        CI = CI - 3                              CCG25        38    A
GRA                ZR     X5,AAV5     IF [CI] = 0   */ END OF TABLE            CCG25        39    A
GRA                BX6    -X0+X5                                               CCG25        40    A
GRA                NZ     X6,AAV2     IF ^( NOCC = 2 & AC & DEF & USE )        CCG25        41    A
GRA                LX5    59-CA.RAP                                            CCG25        42    A
GRA                MI     X5,AAV2     IF RA[CI]     */ PREVIOUSLY ASSIGNED     CCG25        43    A
GRA                SA4    A5+B1       C2W = CI + 1                             CCG25        44    A
GRA                LX4    -IH.RFP                                              CCG25        45    A
GRA                SX3    X4                                                   CCG25        46    A
GRA                ZR     X3,AAV2     IF RF[C2W] = 0                           CCG25        47    A
GRA                SA3    B6+X3       CIF = RCT(RF)                            CCG25        48    A
GRA                LX3    59-CA.DEFP                                           CCG25        49    A
GRA                MI     X3,AAV2     IF DEF[CIF]   */ SUBSCRIPT NOT INVARIANT CCG25        50    A
GRA                LX5    1+CA.RAP-CA.APLNP                                    CCG0119       7    A
GRA                MX7    -CA.APLNL                                            CCG0119       8    A
GRA                BX7    -X7*X5                                               CCG0119       9    A
GRA                NZ     X7,AAV2A    IF REFED ON AP/LN NODES                  CCG0119      10    A
GRA                SA2    HO$UO                                                CCG0119      11    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                ZR     X2,AAV2     IF NOT UNSAFE OPT ALLOWED                          CCG0119      12     A
GRA       AAV2A    BSS    0                                                              CCG0119      13     A
GRA                                                                                      CCG25        51     A
GRA       *        CHECK FOR INTERFERING REFS                                             CCG25        52     A
GRA                                                                                      CCG25        53     A
GRA                LX4    IH.RFP                                                         CCG25        54     A
GRA                SB3    A4                                                             CCG25        55     A
GRA                SA2    B6+B4       C2K = RCT(LCT+1)                                   CCG25        56     A
GRA                MX7    -IH.IHL                                                        CCG25        57     A
GRA       AAV3     SA2    A2-3        C2K = C2K - 3                                      CCG25        58     A
GRA                ZR     X2,AAV4     IF [C2K] = 0  */ END OF TABLE                      CCG25        59     A
GRA                BX6    X2-X4                                                          CCG25        60     A
GRA                BX3    -X7*X6                                                         CCG25        61     A
GRA                NZ     X3,AAV3     IF IH[C2I] " IH[C2K]                               CCG25        62     A
GRA                SB2    A2-B3                                                          CCG25        63     A
GRA                ZR     B2,AAV3     IF C2K = C2I  */ SAME ENTRY                        CCG25        64     A
GRA                EQ     AAV2        INTERFERENCE                                       CCG25        65     A
GRA                                                                                      CCG25        66     A
GRA       AAV4     SX6    A5-B6       I = CI - RCT                                       CCG25        67     A
GRA                SA6    AXCT+B5     AXCT(J) = I   */ SAVE ORDINAL                      CCG25        68     A
GRA                SB5    B5+B1       J = J + 1                                          CCG25        69     A
GRA                SX7    B5-3                                                           CCG25        70     A
GRA                MI     X7,AAV2     IF J < 3                                           CCG25        71     A
GRA                                                                                      CCG25        72     A
GRA       AAV5     SA1    MAA                                                            CCG25        73     A
GRA                MX6    0                                                              CCG25        74     A
GRA                SX7    B5                                                             CCG25        75     A
GRA                SA6    AXCT+B5     AXCT(J) = 0                                        CCG25        76     A
GRA                IX6    X1-X7       MAA = MAA - J                                      CCG25        77          I
          -CCG41
GRA                SB3    5                                                   002360     CCG41         5          I
          -CCG0018
GRA                SX5    B3-B5                                               002370     CCG41         6          I
          -CCG0018
GRA                SA2    MRA                                                            CCG0018       5     A
GRA                AX2    21B                                                            CCG0018       6     A
GRA                MX3    -5                                                             CCG0018       7     A
GRA                BX4    -X3*X2                                                         CCG0018       8     A
GRA                CX3    X4          N.LDREG = COUNT(MRA_LOAD_REGISTERS)                CCG0018       9     A
GRA                IX5    X3-X7                                                          CCG0018      10     A
GRA                IX2    X1-X5                                               002380     CCG41         7     A
GRA                AX2    59                                                  002390     CCG41         8     A
GRA                BX1    X2*X1                                               002400     CCG41         9     A
GRA                BX5    -X2*X5                                              002410     CCG41        10     A
GRA                BX6    X1+X5       MAA=MIN(MAA,5-NXA)                      002420     CCG41        11          I
          -CCG0018
GRA                BX6    X1+X5       MAA = MIN( MAA , N.LDREG-NXA )                     CCG0018      11     A
GRA                SA6    A1                                                             CCG25        78          I
          -CCG0057
GRA                SA2    N.LD                                                           CCG0057       7     A
GRA                SB3    X2                                                             CCG0057       8     A
GRA                SB3    B3-B5                                                          CCG0057       9     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SB4    X6                                                  CCG0057      10    A
GRA                LE     B3,B4,AAV5A IF N.LD - #X/A LE MAA                  CCG0057      11    A
GRA                SX6    X6-1        MAA = MAA - 1                          CCG0057      12    A
GRA                                                                          CCG0057      13    A
GRA       AAV5A    SA6    A1                                                  CCG0057      14    A
GRA                ZR     B5,AAV      IF J = 0      */ NO A/X ASSIGNMENTS    CCG25        79    A
GRA                                                                          CCG25        80    A
GRA       *        MARK THE ASSIGNMENTS IN *RCT* AND *RAT*                    CCG25        81    A
GRA                                                                          CCG25        82    A
GRA                SA1    MRA                                                 CCG25        83    A
GRA                SB3    20B         REGN = 20B                             CCG25        84    A
GRA                SB4    B0          K = 0                                  CCG25        85    A
GRA                                                                          CCG25        86    A
GRA       AAV6     SA5    DAAB                                                CCG25        87    A
GRA                SA2    AXCT+B4     I = AXCT(K)                            CCG25        88    A
GRA                SA3    B6+X2       CI = RCT(I)                            CCG25        89    A
GRA                BX6    X5+X3       (RA,VC)[CI] = 1                        CCG25        90    A
GRA                SA6    A3                                                  CCG25        91    A
GRA                MX0    -CA.NOCCL                                          CCG25        92    A
GRA                LX3    -CA.NOCCP                                          CCG25        93    A
GRA                BX7    -X0*X3                                             CCG25        94    A
GRA                SA4    A3+B1       C2I = CI + 1                           CCG25        95    A
GRA                LX4    -IH.RFP     R = RF[C2I]                            CCG25        96    A
GRA                SA5    B6+X4       CF = RCT(R)                            CCG25        97    A
GRA                LX7    CA.SUSEP                                           CCG25        98    A
GRA                IX6    X5-X7       SUSE[CF] = SUSE[CF] - NOCC[CI]         CCG25        99    A
GRA                SA6    A5                                                  CCG25       100    A
GRA                LX6    -CA.SUSEP                                          CCG25       101    A
GRA                BX7    -X0*X6                                             CCG25       102    A
GRA                SX0    B1                                                 CCG25       103    A
GRA                NZ     X7,AAV9     IF SUSE[CF] " 0                        CCG25       104    A
GRA                                                                          CCG25       105    A
GRA       *        VAR HAS NO USES, DEALLOCATE THE REGISTER ASSIGNED TO IT    CCG25       106    A
GRA                                                                          CCG25       107    A
GRA                LX6    CA.SUSEP                                           CCG25       108    A
GRA                MX4    1                                                  CCG25       109    A
GRA                LX4    1+CA.RAP                                           CCG25       110    A
GRA                BX7    X4*X6                                              CCG0117       6    A
GRA                ZR     X7,AAV9     IF NOT RF[CF]   */ NOT IN A REG        CCG0117       7    A
GRA                BX6    -X4*X6      RA[CF] = 0                             CCG25       111    A
GRA                SA6    A6                                                  CCG25       112    A
GRA                SX7    X6                                                  CCG25       113    A
GRA                NZ     X7,AAV8     IF UDI[CF] " 0  */ PROGRAMMER VARIABLE CCG25       114    A
GRA                LX6    59-CA.CONP                                         CCG25       115    A
GRA                MI     X6,AAV8     IF CON[CF]                             CCG25       116    A
GRA                                                                          CCG25       117    A
GRA                SA4    A5+B1                                              CCG25       118    A
GRA                MX7    -T.REGL                                            CCG25       119    A
GRA                LX4    -IH.CAP                                            CCG25       120    A
GRA                SB2    X4          C = CA[RCT(R+1)]                       CCG25       121    A
GRA                SA4    O.TET                                              CCG25       122    A
GRA                LX7    T.REGP                                             CCG25       123    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SA4    X4+B2                                          CCG25        124     A
GRA                BX6    X7*X4       REG[TET(C)] = 0                    CCG25        125     A
GRA                SA6    A4                                             CCG25        126     A
GRA                                                                      CCG25        127     A
GRA       AAV8     SA4    A5+2        CCW = RCT(R+2)                     CCG25        128     A
GRA                MX7    -CC.REG1L                                      CCG25        129     A
GRA                SX6    B0                                             CCG25        130     A
GRA                LX4    -CC.REG1P                                      CCG25        131     A
GRA                BX4    -X7*X4      REG = REG[CCW]                     CCG25        132     A
GRA                SB2    X4                                             CCG25        133     A
GRA                SA6    RAT+X4      RAT(REG) = 0                       CCG25        134     A
GRA                LX7    B2,X0                                          CCG25        135     A
GRA                BX1    X1+X7       MRA = MRA ! SHIFT(1,REG)           CCG25        136     A
GRA                SA4    NAB                                            CCG25        137     A
GRA                SX7    X4+B1       NAB = NAB + 1                      CCG25        138     A
GRA                SA7    A4                                             CCG25        139     A
GRA                                                                      CCG25        140     A
GRA       AAV9     SB3    B3+1        REGN = REGN + 1                    CCG25        141     A
GRA                LX7    B3,X0       BIT = SHIFT(1,REGN)                CCG25        142     A
GRA                BX6    X7*X1                                          CCG25        143     A
GRA                ZR     X6,AAV9     IF MRA & BIT = 0  */ REG NOT AVAIL CCG25        144     A
GRA                                                                      CCG25        145     A
GRA                SA4    A3+2        CCW = RCT(I+2)                     CCG25        146     A
GRA                BX1    -X7*X1      MRA = MRA & ^BIT                   CCG25        147     A
GRA                SX7    X2                                             CCG25        148     A
GRA                SA7    RAT+B3      RAT(REGN) = I                      CCG25        149     A
GRA                SX5    B3                                             CCG25        150     A
GRA                LX2    18                                             CCG25        151     A
GRA                BX6    X2+X5                                          CCG25        152     A
GRA                SA6    A2          AXCT(K) = 24/0,18/I,18/REGN        CCG25        153     A
GRA                LX5    CC.REG1P                                       CCG25        154     A
GRA                BX6    X4+X5       REG1[CCW] = REGN                   CCG25        155     A
GRA                SA6    A4                                             CCG25        156     A
GRA                SB4    B4+B1       K = K + 1                          CCG25        157     A
GRA                LT     B4,B5,AAV6  IF K < NAXA                        CCG25        158     A
GRA                                                                      CCG25        159     A
GRA                BX6    X1                                             CCG25        160     A
GRA                SA6    A1          UPDATE MRA                         CCG25        161     A
GRA                EQ     AAV                                            CCG25        162     A
GRA        MFA     TTL    GRA - GLOBAL REGISTER ASSIGNMENT / FINAL B-ASSIGNMENT  GRA     2713     A
GRA        MFA     TITLE  MFA - MAKE FINAL B-ASSIGNMENTS                 GRA         2714     A
GRA       **       MFA - MAKE FINAL B-ASSIGNMENTS                       GRA         2715     A
GRA       *                                                             GRA         2716     A
GRA       *        ADJUST ADDRESS REFERENCES TO MINIMIZE SIZE OF CODE IN LOOP.  GRA  2717     A
GRA       *        ASSIGN CONSTANTS TO B-REGS, SETUP LOOP COUNTING CODE INFO.  GRA  2718     A
GRA                                                                      GRA         2719     A
GRA       *        BC. - B-REG CANIDATE SORT TABLE                      GRA         2720     A
GRA                                                                      GRA         2721     A
GRA                DESCRIBE BC.,60    BCT(RF,IH,NOCC,CO)                GRA         2722     A
GRA        RF      DEFINE 15          RCT RF                            GRA         2723     A
GRA        IH      DEFINE 18          IH OF SYMBOL                      GRA         2724     A
GRA        NOCC    DEFINE 9                                             GRA         2725     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA        CO       DEFINE 18          RCT INDEX OF CANIDATE                          GRA        2726     A
GRA                                                                                   GRA        2727     A
GRA       *         AI. - RF CLASS INDEX TABLE                                         GRA        2728     A
GRA                                                                                   GRA        2729     A
GRA                  DESCRIBE AI.,60    AIT(BASE,NIH,LEN,INDX)                         GRA        2730     A
GRA                  DEFINE 2                                                          GRA        2731     A
GRA        BASE      DEFINE 1           =1 IF PROG VAR OR BASE MEMBER OF *IP* CLASS    GRA        2732     A
GRA        NIH       DEFINE 18          377777B - N.IH"S IN RF CLASS                   GRA        2733     A
GRA        LEN       DEFINE 18          NUMBER OF ENTRIES IN CLASS                     GRA        2734     A
GRA                  DEFINE 2                                                          GRA        2735     A
GRA        FP        DEFINE 1                                                          GRA        2736     A
GRA        INDX      DEFINE 18          INDEX TO LWA OF CLASS                          GRA        2737     A
GRA                                                                                   GRA        2738     A
GRA        MFA       ROUTINE                                                           GRA        2739     A
GRA                  SA1    MRA                                                        GRA        2740     A
GRA                  SA5    L.RCT                                                      GRA        2741     A
GRA                  SX2    376B                                                       GRA        2742     A
GRA                  BX3    X2*X1                                                      GRA        2743     A
GRA                  SX4    X3+B1                                                      GRA        2744     A
GRA                  PX6    X4         ABR = PACK( 0 , MRA&376B + 1 )                  GRA        2745     A
GRA                  SA6    ABR                                                        GRA        2746     A
GRA                  ALLOC  RND,X5     ALLOC( RND , L.RCT )                            GRA        2747     A
GRA                  SA4    O.RCT                                                      GRA        2748     A
GRA                  MX7    0                                                          GRA        2749     A
GRA                  SB7    X2+B1      BO = O.RND + 1                                  GRA        2750     A
GRA                  SA7    X2         BI = BO - 1;  [BI] = 0                          GRA        2751     A
GRA                  SB6    X4         CO = [O.RCT]                                    GRA        2752     A
GRA                  SA5    B6+X5      CI = CO + L.RCT                                 GRA        2753     A
GRA                  MX0    -CA.NOCCL                                                  GRA        2754     A
GRA                  SB4    B0         NC = 0        */ N. CONS                        GRA        2755     A
GRA                  LX0    CA.NOCCP                                                   GRA        2756     A
GRA                  SB5    B0         LC = 0        */ LINK TO LAST CON               GRA        2757     A
GRA                  MX6    60                                                         GRA        2758     A
GRA                  SA6    B6         [CO] = -0     */ INDICATE RF=0 IS IN A REG      GRA        2759     A
GRA                  EQ     MFA2                                                       GRA        2760     A
GRA                                                                                   GRA        2761     A
GRA       *         ADD ENTRY TO *CON* LIST                                            GRA        2762     A
GRA                                                                                   GRA        2763     A
GRA        MFA1      SA3    A5+B1                                                      GRA        2764     A
GRA                  SA4    A3+B1                                                      GRA        2765     A
GRA                  LX3    59-IH.MSKP                                                 GRA        2766     A
GRA                  MI     X3,MFA2    IF MSK[CI+1]                                    GRA        2767     A
GRA                  SX7    B5                                                         GRA        2768     A
GRA                  SB4    B4+B1      NC = NC + 1                                     GRA        2769     A
GRA                  LX7    CC.INCP    INC[CI+2] = LC                                  GRA        2770     A
GRA                  BX6    X4+X7                                                      GRA        2771     A
GRA                  SB5    A5-B6      LC = CI - CO  */ POINT TO LAST                  GRA        2772     A
GRA                  SA6    A4                                                         GRA        2773     A
GRA                                                                                   GRA        2774     A
GRA       *         SCAN *RCT* AND BUILD *CON* LIST AND RF,IH SORT TABLE               GRA        2775     A
GRA       *         OF ADDRESS CANIDATES                                               GRA        2776     A
GRA                                                                                   GRA        2777     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA       MFA2     SA5    A5-3        CI = CI - 3                                  GRA          2778     A
GRA                ZR     X5,MFA3     IF [CI] = 0   */ END OF TABLE                GRA          2779     A
GRA                BX6    X5                                                       GRA          2780     A
GRA                LX6    59-CA.CONP                                               GRA          2781     A
GRA                MI     X6,MFA1     IF CON[CI]                                   GRA          2782     A
GRA                LX6    CA.CONP-CA.RAP                                           GRA          2783     A
GRA                MI     X6,MFA2     IF RA[CI]                                    GRA          2784     A
GRA                LX6    CA.RAP-CA.ACP                                            GRA          2785     A
GRA                MI     X6,MFA2A    IF AC[CI]                                    GRA          2786          I
          -FCC1531X
GRA                LX6    CA.ACP-CA.FPP                                            GRA          2787          I
          -FCC1531X
GRA                PL     X6,MFA2     IF ^FP[CI]                                   GRA          2788          I
          -FCC1531X
GRA                LX6    CA.FPP-CA.RFP                                            GRA          2789          I
          -FCC1014  -FCC1531X
GRA                MI     X6,MFA2     IF RF[CAW]                                   GRA          2790          I
          -FCC1014  -FCC1531X
GRA                                                                               GRA          2791          I
          -FCC1531X
GRA                SX1    B1                                                       GRA          2792          I
          -FCC1531X
GRA                LX1    CA.ACP                                                   GRA          2793          I
          -FCC1531X
GRA                BX6    X5+X1       AC[CI] = 1                                   GRA          2794          I
          -FCC1531X
GRA                LX1    CA.VCP-CA.ACP                                            GRA          2795          I
          -FCC1531X
GRA                BX6    -X1*X6      VC[CI] = 0                                   GRA          2796          I
          -FCC1531X
GRA                SA6    A5                                                       GRA          2797          I
          -FCC1531X
GRA                                                                               GRA          2798          I
          -FCC1531X
GRA       MFA2A    SA4    A5+B1       C2W = [CI+1]                                 GRA          2799          I
          -FCC1531X
GRA                PL     X6,MFA2     IF ^AC[CI]                                   FCC1531X        5     A
GRA                                                                               FCC1531X        6     A
GRA                SA4    A5+2        CCW = [RCT+2]                                FCC1828X        6     A
GRA                LX4    -CC.H2P                                                  FCC1828X        7     A
GRA                SX3    X4                                                       FCC1828X        8     A
GRA                NZ     X3,MFA2     IF H2[CCW] " 0  */ OMIT SYM DIFFERENCES      FCC1828X        9     A
GRA                SA4    A5+B1       C2W = [CI+1]                                 FCC1531X        7     A
GRA                BX5    -X0*X5                                                   GRA          2800     A
GRA                SX3    X4                                                       GRA          2801     A
GRA                LX4    -IH.RFP                                                  GRA          2802     A
GRA                SX4    X4          R = RF[CI]                                   GRA          2803     A
GRA                LX5    BC.NOCCP-CA.NOCCP                                        GRA          2804     A
GRA                SA2    B6+X4       CJ = RCT(R)                                  GRA          2805          I
          -FCC1014
GRA                LX2    59-CA.RAP                                                GRA          2806          I
          -FCC1014

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                PL     X2,MFA2      IF ^RA[CJ]                                  GRA        2807          I
          -FCC1014
GRA                LX3    BC.IHP                                                   GRA        2808          I
          -FCC1014
GRA                ZR     X4,MFA2B     IF R = 0                                    FCC1014      83     A
GRA                                                                                FCC1014      84     A
GRA                SA1    DAAB                                                     FCC1014      85     A
GRA                SA2    B6+X4        CJ = RCT(R)                                 FCC1014      86     A
GRA                BX6    -X1+X2                                                   FCC1014      87     A
GRA                NZ     X6,MFA2B     IF ^( VC[CJ] & RA[CJ] )                     FCC1014      88     A
GRA                SA2    A2+2         CCW = RCT(R+2)                              FCC1014      89     A
GRA                LX2    59-CC.REG1P-4                                            FCC1014      90     A
GRA                PL     X2,MFA2C     IF REG1[CCW] < 20B  */ IN A B-REGISTER      FCC1014      91     A
GRA                                                                                FCC1014      92     A
GRA       MFA2B    ZR     X3,MFA2      IF IH[C2W] = 0                              FCC1014      93     A
GRA                MX4    0            R = 0                                       FCC1014      94     A
GRA                SA2    A5                                                       FCC1014      95     A
GRA                SA1    MFAD                                                     FCC1014      96     A
GRA                BX6    X1*X2                                                    FCC1014      97     A
GRA                NZ     X6,MFA2C     IF FP[CI] ! NOCC[CI] \ 2                    FCC1014      98     A
GRA                LX2    59-CA.RAP                                                FCC1014      99     A
GRA                PL     X2,MFA2      IF ^RA[CI]                                  FCC1014     100     A
GRA                                                                                FCC1014     101     A
GRA       MFA2C    LX3    BC.IHP                                                   FCC1014     102     A
GRA                SX6    A5-B6                                                    GRA        2809     A
GRA                LX4    BC.RFP                                                   GRA        2810     A
GRA                BX3    X3+X4                                                    GRA        2811     A
GRA                IX5    X5+X6                                                    GRA        2812     A
GRA                BX7    X3+X5                                                    GRA        2813     A
GRA                SA7    A7+B1        BI = BI + 1;  [BI] = BCT(R,IH,NOCC,CI-CO)   GRA        2814     A
GRA                EQ     MFA2                                                     GRA        2815     A
GRA                                                                                GRA        2816     A
GRA       MFA3     SX6    B5                                                       GRA        2817     A
GRA                SA6    CL           CL = LC       */ CON LIST POINTER           GRA        2818     A
GRA                MX6    0                                                        GRA        2819     A
GRA                SA6    B6           [CO] = 0                                    GRA        2820     A
GRA                SB3    A7+B1        BI = BI + 1                                 GRA        2821     A
GRA                SX1    B3-B7        LEN = BI - BO                               GRA        2822     A
GRA                BX6    X1                                                       GRA        2823     A
GRA                SA6    NAC          NAC = LEN     */ NUMBER OF ADDRESS CANIDATE GRA        2824     A
GRA                ZR     X1,MFA48     IF LEN = 0                                  GRA        2825     A
GRA                                                                                GRA        2826     A
GRA                CALL   SHL          SORT( BCT )                                 GRA        2827          I
          -CCG1
GRA                CALL   SHL=         SORT( BCT )                                 CCG1        112          I
          -CCGNC
GRA                CALL   SHL#         SORT( BCT )                                 CCGNC       654     A
GRA                                                                                GRA        2828     A
GRA       *        NOW FORM THE RF CLASS INDEX TABLE AND SORT IT                   GRA        2829     A
GRA                                                                                GRA        2830     A
GRA                SA2    O.TET                                                    GRA        2831     A
GRA                MX7    60                                                       GRA        2832     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA              SA7    B3          AI = BI;  [AI] = -0  */ SEARCH TERMINATOR          GRA        2833    A
GRA              MX0    -BC.RFP                                                        GRA        2834    A
GRA              SB3    B3+B1       AF = AI + 1   */ BASE                              GRA        2835    A
GRA              SA0    X2                                                             GRA        2836    A
GRA              SB4    377776B     NIH = -1                                           GRA        2837    A
GRA              MX1    -BC.IHL                                                        GRA        2838    A
GRA              SB5    B7          BF = BO       */ ADDRESS OF FIRST IN CLASS         GRA        2839    A
GRA              LX1    BC.IHP                                                         GRA        2840    A
GRA              SA5    B7          BI = BO                                            GRA        2841    A
GRA              EQ     MFA6                                                           GRA        2842    A
GRA                                                                                    GRA        2843    A
GRA      *       END OF RF CLASS, SETUP INDEX WORD IN *AIT*                            GRA        2844    A
GRA                                                                                    GRA        2845    A
GRA       MFA4   AX2    BC.RFP                                                         GRA        2846    A
GRA              SA4    B6+X2       CI = RCT(RF[BL])                                   GRA        2847    A
GRA              SB2    X2                                                   000250    FCC2092       1    A
GRA              MX7    0           BASE = 0                                 000260    FCC2092       2    A
GRA              ZR     B2,MFA5     IF RF[BL] = 0                            000270    FCC2092       3    A
GRA              SX7    B1          BASE = 1                                           GRA        2848    A
GRA              SB2    X4                                                             GRA        2849    A
GRA              NZ     B2,MFA5     IF UDI[CI] " 0  */ PROGRAMMER VAR                  GRA        2850    A
GRA                                                                                    GRA        2851    A
GRA              SA4    A4+B1                                                          GRA        2852    A
GRA              LX4    -IH.CAP                                                        GRA        2853    A
GRA              SB2    X4                                                             GRA        2854    A
GRA              SA3    A0+B2       TI = TET(CA[CI+1])                                 GRA        2855    A
GRA              LX3    -T.BIPP                                                        GRA        2856    A
GRA              BX7    X7*X3       BASE = BIP[TI]                                     GRA        2857    A
GRA                                                                                    GRA        2858    A
GRA       MFA5   SX2    A5-B7       INDX = BI - BO                                     GRA        2859    A
GRA              LX2    AI.INDXP                                                       GRA        2860    A
GRA              SX3    A5-B5       LEN = BI - BF                                      GRA        2861    A
GRA              SB5    A5          BF = BI                                            GRA        2862    A
GRA              LX3    AI.LENP                                                        GRA        2863    A
GRA              SX4    B4                                                             GRA        2864    A
GRA              LX7    AI.BASEP                                                       GRA        2865    A
GRA              BX6    X7+X2                                                          GRA        2866    A
GRA              LX4    AI.NIHP                                                        GRA        2867    A
GRA              IX3    X3+X4                                                          GRA        2868    A
GRA              BX7    X3+X6                                                          GRA        2869    A
GRA              SA7    A7+1        AI = AI + 1;  [AI] = AIT(BASE,NIH,LEN,INDX)        GRA        2870    A
GRA              SB4    377776B     NIH = -1                                           GRA        2871    A
GRA              ZR     X5,MFA7     IF [BI] = 0   */ END OF TABLE                      GRA        2872    A
GRA                                                                                    GRA        2873    A
GRA       MFA6   BX2    X5          BL = [BI]                                          GRA        2874    A
GRA              SA5    A5+B1       BI = BI + 1                                        GRA        2875    A
GRA              BX6    X5-X2                                                          GRA        2876    A
GRA              BX7    X0*X6                                                          GRA        2877    A
GRA              NZ     X7,MFA4     IF RF[BI] " RF[BL]                                 GRA        2878    A
GRA                                                                                    GRA        2879    A
GRA              BX6    -X1*X6                                                         GRA        2880    A
GRA              ZR     X6,MFA6     IF IH[BI] = IH[BL]                                 GRA        2881    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SB4    B4-1        NIH = NIH - 1                          GRA        2882     A
GRA                NZ     X5,MFA6     IF [BI] " 0                            GRA        2883     A
GRA                                                                          GRA        2884     A
GRA                EQ     MFA5                                               GRA        2885     A
GRA                                                                          GRA        2886     A
GRA      *         SORT THE CLASS INDEX TABLE                                GRA        2887     A
GRA                                                                          GRA        2888     A
GRA       MFA7     SB2    A7+B1                                              GRA        2889     A
GRA                SB7    B7-B1                                              GRA        2890     A
GRA                SX6    B2-B7       ATI = AI+1 - (BO-1)                    GRA        2891     A
GRA                MX7    0                                                  GRA        2892     A
GRA                SA7    A5          [BI] = 0                               GRA        2893     A
GRA                SA6    ATI                                                GRA        2894     A
GRA                SA6    L.RND                                              GRA        2895     A
GRA                SX1    B2-B3       LEN = AI+1 - AF                        GRA        2896     A
GRA                SB7    B3                                                 GRA        2897     A
GRA                CALL   SHL         SORT( AIT )                            GRA        2898          I
         -CCG1
GRA                CALL   SHL=        SORT( AIT )                            CCG1        113          I
         -CCGNC
GRA                CALL   SHL#        SORT( AIT )                            CCGNC       655     A
GRA                                                                          GRA        2899     A
GRA                PRINT  MFA,(* NAC,ATI =*2Z5),(NAC,ATI)                    GRA        2900     A
GRA                TRACE  MFA,RND                                             GRA        2901     A
GRA                                                                          GRA        2902     A
GRA      *         GET SPACE FOR NEW ENTRIES ( CON"S, DIFFERENCES )          GRA        2903     A
GRA                                                                          GRA        2904     A
GRA                SA5    NAC                                                GRA        2905     A
GRA                LX4    B1,X5                                              GRA        2906     A
GRA                IX1    X4+X5                                              GRA        2907     A
GRA                ALLOC  RCT,X1      ALLOC( RCT , 3*NAC )                   GRA        2908     A
GRA                SA4    O.RND                                              GRA        2909     A
GRA                SX6    B6                                                 GRA        2910     A
GRA                SB7    X4                                                 GRA        2911     A
GRA                SA6    L.RCT       L.RCT = OLD(L.RCT)                     GRA        2912     A
GRA                SB6    X2                                                 GRA        2913     A
GRA       MFA      SPACE  2,10                                               GRA        2914     A
GRA      **        MAIN LOOP FOR ADDRESS CANIDATE MODIFICATION               GRA        2915     A
GRA      *         (A0) = REGNO,  (B6) = [O.RCT],  (B7) = [O.RND]            GRA        2916     A
GRA                                                                          GRA        2917     A
GRA       MFA10    SA1    ATI                                                GRA        2918     A
GRA                SX7    X1-1        ATI = ATI - 1                          GRA        2919     A
GRA                SA2    B7+X7       AI = RND(ATI)                          GRA        2920     A
GRA                SA7    A1                                                 GRA        2921     A
GRA                ZR     X2,MFA48    IF AI = 0     */ END OF TABLE          GRA        2922     A
GRA                                                                          GRA        2923     A
GRA                LX2    -AI.INDXP                                          GRA        2924     A
GRA                SA5    B7+X2       BI = RND(INDX[AI])                     GRA        2925     A
GRA                LX2    AI.INDXP-AI.LENP                                   GRA        2926     A
GRA                SB5    X2          L = LEN[AI]   */ NUMBER OF ENTRIES     GRA        2927     A
GRA                LX2    AI.LENP-AI.NIHP                                    GRA        2928     A
GRA                SB2    X2+400000B                                         GRA        2929     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                LX2    AI.NIHP+59-AI.BASEP                                  GRA        2930     A
GRA                BX6    X5                                                   GRA        2931     A
GRA                SB4    -B2          N = -NIH[AI]  */ NUMBER OF IH"S         GRA        2932     A
GRA                AX6    BC.RFP                                               GRA        2933     A
GRA                SB2    X6+2         R = RF[BI]                              GRA        2934     A
GRA                MX0    -CC.REG1L                                            GRA        2935     A
GRA                SA3    B6+B2        CCW = RCT(R+2)                          GRA        2936     A
GRA                LX3    -CC.REG1P                                            GRA        2937     A
GRA                BX7    -X0*X3                                               GRA        2938     A
GRA                SA0    X7           REGNO = REG1[CCW]                       GRA        2939     A
GRA                ZR     X7,MFA40     IF REGNO = 0                            GRA        2940     A
GRA                                                                            GRA        2941     A
GRA                SA4    A3-2         CAW = RCT(R)                            GRA        2942     A
GRA                SA1    MTAB                                                 GRA        2943     A
GRA                BX6    X1*X4                                                GRA        2944     A
GRA                LX4    59-CA.TUP                                            GRA        2945     A
GRA                NZ     X6,MFA40     IF (IRA,EU,LX)[CAW]  */ CANT BIAS       GRA        2946     A
GRA                PL     X4,MFA10A    IF ^TU[CAW]   */ NOT LOOP *CV* OR LIMIT GRA        2947     A
GRA                SA1    TRD                                                  GRA        2948     A
GRA                SX4    X1-2                                                 GRA        2949     A
GRA                ZR     X4,MFA40     IF TRD = 2    */ I,N IN REGS            GRA        2950     A
GRA                                                                            GRA        2951     A
GRA       MFA10A   GT     B5,B1,MFA20 IF L > 1      */ MORE THAN 1 MEMBER IN CLASS GRA    2952     A
GRA                                                                            GRA        2953     A
GRA      *         1 ENTRY IN RF,IH CLASS, SETUP ADDRESS IN B-REG WITH RF      GRA        2954     A
GRA                                                                            GRA        2955     A
GRA                PL     X2,MFA12     IF ^BASE[AI]                            GRA        2956     A
GRA                                                                            GRA        2957     A
GRA      *         ASSIGN BASE MEMBER ADDRESS TO B-REG                         GRA        2958     A
GRA                                                                            GRA        2959     A
GRA       MFA11    RJ     AFA          ASSIGN FULL ADDRESS                     GRA        2960     A
GRA                EQ     MFA10                                                GRA        2961     A
GRA                                                                            GRA        2962     A
GRA      *         ^BASE *IP*, FIND BASE TP IN REGS AND DO DIFFERENCING        GRA        2963     A
GRA                                                                            GRA        2964     A
GRA       MFA12    SA4    A4+B1                                                GRA        2965     A
GRA                SA3    O.TET                                                GRA        2966     A
GRA                LX4    -IH.CAP                                              GRA        2967     A
GRA                SB2    X4           C = CA[RCT(R+1)]  */ CA OF *IP*         GRA        2968     A
GRA                SA2    X3+B2                                                GRA        2969     A
GRA                SA1    O.IIT                                                GRA        2970     A
GRA                LX2    -T.ITIP      ITI = ITI[TET(C)]  */ INDX TO INFO WORD GRA        2971          I
         -CCGA063
GRA                LX2    59-T.INVP                                            CCGA063       6     A
GRA                MI     X2,MFA11     IF INV[TET(C)]                          CCGA063       7     A
GRA                LX2    1+T.INVP-T.ITIP                                      CCGA063       8     A
GRA                SB2    X2                                                   GRA        2972     A
GRA                ZR     B2,MFA11     IF ITI = 0    */ INVARAINT TEMP         GRA        2973     A
GRA                SA4    X1+B2        ITW = IIT(ITI)                          GRA        2974     A
GRA                LX4    -PI.CAP                                              GRA        2975     A
GRA                SB2    X4           BC = CA[ITW]  */ CA OF BASE             GRA        2976     A
GRA                SA2    X3+B2                                                GRA        2977     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                LX2    -T.REGP                                                GRA          2978     A
GRA                SB5    X2          BREGN = REG[TET(BC)]  */ REG THAT BASE IS I GRA          2979     A
GRA                ZR     B5,MFA11    IF BREGNO = 0 */ BASE NOT IN A REG         GRA          2980     A
GRA                SA1    RAT+B5      J1 = R1[RAT(BREGN)]                        FCC1518X       19     A
GRA                SA2    B6+X1                                                  FCC1518X       20     A
GRA                LX2    59-CA.EUP                                              FCC1518X       21     A
GRA                MI     X2,MFA11    IF EU[RCT(J1)]  */ BASE IP IS *XU*         FCC1518X       22     A
GRA                                                                             GRA          2981     A
GRA      *         NOW PLAY AROUND TO SETUP *REGNO* AS DELTA(IHCA,RF"S)          GRA          2982     A
GRA                                                                             GRA          2983     A
GRA                SA1    B6+X5       J = CO[BI]                                 GRA          2984     A
GRA                MX0    1                                                      GRA          2985     A
GRA                SA2    A1+2        CCW = RCT(J+2)                             GRA          2986     A
GRA                LX0    1+CA.RAP                                               GRA          2987     A
GRA                SX3    B5+2S6      REG1[CCW] = BREGN                          GRA          2988          I
          -FCC0977
GRA                LX3    CC.REG1P    MIT[CCW] = 2                               GRA          2989          I
          -FCC0977
GRA                SX3    B5+2S6+V.SRF  REG1[CCW] = BREGN                        FCC0977        11     A
GRA                LX3    CC.REG1P    MIT[CCW] = 42B    */ SET *SRF* BIT         FCC0977        12     A
GRA                SX4    A0          REG2[CCW] = REGNO                          GRA          2990     A
GRA                LX4    CC.REG2P                                               GRA          2991     A
GRA                BX6    X0+X1       RA[RCT(J)] = 1                             GRA          2992     A
GRA                SA6    A1                                                     GRA          2993     A
GRA                BX3    X3+X4                                                  GRA          2994     A
GRA                BX7    X3+X2                                                  GRA          2995     A
GRA                SA7    A2                                                     GRA          2996     A
GRA                SA7    GRASRF      GRASRF = 1                                 FCC0977        13     A
GRA                                                                             GRA          2997     A
GRA      *         SETUP *RAT* TO FORM DIFFERENCE CODE IN PROLOGUE               GRA          2998     A
GRA                                                                             GRA          2999     A
GRA                SA1    RAT+B5                                                 GRA          3000     A
GRA                SX3    X1          J1 = R1[RAT(BREGN)]                        GRA          3001     A
GRA                LX1    -RA.R3P                                                GRA          3002     A
GRA                SB3    X1          J3 = R3[RAT(BREGN)]                        GRA          3003     A
GRA                SA2    RAT+A0      JO = R1[RAT(REGNO)]                        GRA          3004     A
GRA                LX0    RA.INVP-CA.RAP                                         GRA          3005     A
GRA                BX2    X0+X2       INV[RAT(REGNO)] = 1                        GRA          3006     A
GRA                LX3    RA.R2P                                                 GRA          3007     A
GRA                BX7    X3+X2       R2[RAT(REGNO)] = J1                        GRA          3008     A
GRA                SA7    A2                                                     GRA          3009     A
GRA                SA4    SEEA                                                   GRA          3010     A
GRA                SA3    B6+X2                                                  GRA          3011     A
GRA                BX6    X3+X4       KD[RCT(JO)] = 1                            GRA          3012     A
GRA                SA6    A3                                                     GRA          3013     A
GRA                NZ     B3,MFA13    IF J3 " 0     */ SYMBOL APPENDED           GRA          3014     A
GRA                                                                             GRA          3015     A
GRA                SX3    X5                                                     GRA          3016     A
GRA                LX3    RA.R3P                                                 GRA          3017     A
GRA                BX7    X3+X7       R3[RAT(REGNO)] = J  */ APPEND THIS IHCA    GRA          3018     A
GRA                SA7    A7                                                     GRA          3019     A
GRA                EQ     MFA10                                                  GRA          3020     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                                                                                GRA      3021   A
GRA      *         BASE REG HAS A SYMBOL APPENDED, FORM ADDRESS DIFFERENCE         GRA      3022   A
GRA                                                                                GRA      3023   A
GRA       MFA13    SX5    X5+B1                                                    GRA      3024   A
GRA               SA1    B6+X5                                                     GRA      3025   A
GRA               SB3    B3+B1                                                     GRA      3026   A
GRA               SA2    B6+B3                                                     GRA      3027   A
GRA               MX0    -IH.RFP                                                   GRA      3028   A
GRA               BX1    -X0*X1      CAIH1 = CAIH[RCT(J+1)]                        GRA      3029   A
GRA               BX2    -X0*X2      CAIH2 = CAIH[RCT(J3+1)]                       GRA      3030   A
GRA               BX3    X1-X2                                                     GRA      3031   A
GRA               ZR     X3,MFA10    IF CAIH1 = CAIH2                              GRA      3032   A
GRA                                                                                GRA      3033   A
GRA               SA4    L.RCT                                                     GRA      3034   A
GRA               SB2    X3                                                        GRA      3035   A
GRA               SB4    X4+B1       OL = L.RCT                                    GRA      3036   A
GRA               SX6    X4+3        L.RCT = L.RCT + 3                             GRA      3037   A
GRA               LX4    RA.R3P                                                    GRA      3038   A
GRA               SA6    A4                                                        GRA      3039   A
GRA               BX7    X4+X7       R3[RAT(REGNO)] = OL  */ DIFF CANIDATE         GRA      3040   A
GRA               MX0    -IH.CAL                                                   GRA      3041   A
GRA               SA7    A7                                                        GRA      3042   A
GRA               SX3    X1                                                        GRA      3043   A
GRA               SX4    X2                                                        GRA      3044   A
GRA               LX1    -IH.CAP                                                   GRA      3045   A
GRA               LX2    -IH.CAP                                                   GRA      3046   A
GRA               SX1    X1                                                        GRA      3047   A
GRA               SX2    X2                                                        GRA      3048   A
GRA               IX1    X1-X2                                                     GRA      3049   A
GRA               MX7    0           KEY2 = 0                                      GRA      3050   A
GRA               BX6    -X0*X1                                                    GRA      3051   A
GRA               LX6    IH.CAP      KEY = IHW(0,0, CA1-CA2 , 0 )                  GRA      3052   A
GRA               ZR     B2,MFA14    IF IH1 = IH2                                  GRA      3053   A
GRA               SX7    X4                                                        GRA      3054   A
GRA               LX7    CC.H2P      KEY2 = CCW(0,IH2,0)                           GRA      3055   A
GRA               BX6    X3+X6       IH[KEY] = IH1                                 GRA      3056   A
GRA                                                                                GRA      3057   A
GRA       MFA14   SA4    EDCA                                                      GRA      3058   A
GRA               SA6    B6+B4       RCT(OL+1) = KEY                               GRA      3059   A
GRA               SA7    A6+B1       RCT(OL+2) = KEY2                              GRA      3060   A
GRA               BX6    X4                                                        GRA      3061   A
GRA               SA6    A6-B1       RCT(OL) = CAW(0,AC,PL)                        GRA      3062   A
GRA               EQ     MFA10                                                     GRA      3063   A
GRA       MFA     SPACE  2,10                                                      GRA      3064   A
GRA      *        PROCESS CLASS WITH L > 1 AND NIH = 1                             GRA      3065   A
GRA                                                                                GRA      3066   A
GRA       MFA20   NE     B4,B1,MFA30 IF N > 1                                      GRA      3067   A
GRA               RJ     PSC         PROCESS SIMPLE CLASS                          GRA      3068   A
GRA               EQ     MFA10                                                     GRA      3069   A
GRA       MFA     SPACE  2,10                                                      GRA      3070   A
GRA      *        MULTI IH CLASS                                                   GRA      3071   A
GRA                                                                                GRA      3072   A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA       MFA30    SX6    B4-5                                                GRA        3073    A
GRA                PL     X6,MFA40    IF N > 4                                GRA        3074    A
GRA                SA4    NAB                                                 GRA        3075    A
GRA                ZR     X4,MFA10    IF NAB = 0                              GRA        3076    A
GRA                                                                           GRA        3077    A
GRA      *         1 < NIH @ 4 , TRY ADDRESS DIFFERENCING                     GRA        3078    A
GRA      *         FIRST FORM AN *IH* CLASS SUBTABLE                          GRA        3079    A
GRA                                                                           GRA        3080    A
GRA                SB3    B0          ML = 0        */ MAX *IH* CLASS LEN    GRA        3081    A
GRA                MX0    -BC.IHP                                             GRA        3082    A
GRA                SX1    B0          NC = 0                                  GRA        3083    A
GRA                SX7    A5          BF = BI                                 GRA        3084    A
GRA                                                                           GRA        3085    A
GRA       MFA31    BX4    X5          BL = [BI]                               GRA        3086    A
GRA                SA5    A5-B1       BI = BI - 1                             GRA        3087    A
GRA                BX6    X4-X5                                               GRA        3088    A
GRA                BX3    X0*X6                                               GRA        3089    A
GRA                ZR     X3,MFA31    IF IH[BI] = IH[BL]                      GRA        3090         I
          -FCC0814
GRA                ZR     X3,MFA31    IF RFIH[BI] = RFIH[BL]                  FCC0814       5    A
GRA                                                                           GRA        3091    A
GRA                SA2    B6+X4       CI = RCT(CO[BL])                        GRA        3092    A
GRA                SX3    B1                                                  GRA        3093    A
GRA                LX3    CA.FPP                                              GRA        3094    A
GRA                BX3    X3*X2                                               GRA        3095    A
GRA                SX2    A5                                                  GRA        3096    A
GRA                IX6    X7-X2       LEN = BF - BI                           GRA        3097    A
GRA                LX3    AI.FPP-CA.FPP     FP = FP[CI]                       GRA        3098    A
GRA                SB2    X6                                                  GRA        3099    A
GRA                BX7    X3+X7                                               GRA        3100    A
GRA                LX6    AI.LENP                                             GRA        3101    A
GRA                BX7    X6+X7                                               GRA        3102    A
GRA                SA7    CIT+X1      CIT(NC) = AIT(0,0,LEN,FP,BF)            GRA        3103    A
GRA                SX1    X1+B1       NC = NC + 1                             GRA        3104    A
GRA                LE     B2,B3,MFA32 ML = MAX( ML , LEN )                    GRA        3105    A
GRA                SB3    B2                                                  GRA        3106    A
GRA       MFA32    SX7    A5                                                  GRA        3107    A
GRA                SB2    X1                                                  GRA        3108    A
GRA                LT     B2,B4,MFA31 IF NC < N                               GRA        3109    A
GRA                                                                           GRA        3110    A
GRA                SA2    NAB                                                 GRA        3111    A
GRA                SA3    CIT                                                 GRA        3112    A
GRA                SB2    X2+B3                                               GRA        3113    A
GRA                SA5    X3          BI = INDX[CIT]                          GRA        3114    A
GRA                GT     B5,B2,MFA40 IF L > NAB + ML  */ NOT ENOUGH REGISTERS GRA       3115    A
GRA       MFA      SPACE  2,10                                                GRA        3116    A
GRA      *         BASE *IH* CLASS IS THAT WITH *ML* AND F.P. ( IF ANY )      GRA        3117    A
GRA                                                                           GRA        3118    A
GRA                MX7    0                                                   FCC0814       6    A
GRA                SA7    A7+B1       CIT(NC) = 0   */ TABLE TERMINATOR       FCC0814       7    A
GRA                SB5    B1          I = 1                                   GRA        3119    A
GRA                SB2    B0          IM = 0;  MVAL = CIT(0)                  GRA        3120    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                                                                    GRA      3121    A
GRA       MFA33   SA4    CIT+B5                                        GRA      3122    A
GRA               IX6    X3-X4                                         GRA      3123    A
GRA               SB5    B5+B1        I = I + 1                        GRA      3124    A
GRA               PL     X6,MFA34     IF CIT(I-1) @ MVAL               GRA      3125    A
GRA               SB2    B5-B1        IM = I - 1                       GRA      3126    A
GRA               BX3    X4           MVAL = CIT(IM)                   GRA      3127    A
GRA       MFA34   LT     B5,B4,MFA33 IF I < N                          GRA      3128    A
GRA                                                                    GRA      3129    A
GRA               GT     B3,B1,MFA34A      IF ML > 1                    GRA      3130    A
GRA               LX3    59-AI.FPP                                     GRA      3131    A
GRA               PL     X3,MFA40     IF ^FP[MVAL]  */ NO F.P. S IN THE CLASS  GRA      3132    A
GRA                                                                    GRA      3133    A
GRA       MFA34A  SA2    CIT+B2                                        GRA      3134    A
GRA               SA3    A3                                            GRA      3135    A
GRA               BX6    X3                                            GRA      3136    A
GRA               LX7    X2                                            GRA      3137    A
GRA               SA6    A2           SWAP( CIT(0) , CIT(IM) )         GRA      3138    A
GRA               SA7    A3                                            GRA      3139    A
GRA                                                                    GRA      3140    A
GRA               SA5    X2           BI = INDX[CIT]                   GRA      3141    A
GRA               LX2    -AI.LENP                                      GRA      3142    A
GRA               SB5    X2           L = LEN[CIT]                     GRA      3143    A
GRA               RJ     PSC          PROCESS BASE CLASS               GRA      3144    A
GRA                                                                    GRA      3145    A
GRA               SX6    B1+B1                                         GRA      3146    A
GRA               SA6    CIT          I = 2;  AI = CIT(1)              GRA      3147    A
GRA                                                                    GRA      3148    A
GRA       MFA35   SA1    CIT+1                                         GRA      3149    A
GRA               SA2    MFAC                                          GRA      3150    A
GRA               SA3    X1           BI = INDX[AI]                    GRA      3151    A
GRA               IX6    X1-X2        (LEN,INDX)[AI] = (LEN,INDX)[AI] - 1  GRA      3152    A
GRA               SA6    A1                                            GRA      3153    A
GRA               SB5    X3           J = CO[BI]                       GRA      3154    A
GRA               RJ     AAD          ASSIGN ADDRESS DIFFERENCE( J, REGNO , CAIH  GRA      3155    A
GRA               SA1    CIT+1                                         GRA      3156    A
GRA               LX1    -AI.LENP                                      GRA      3157    A
GRA               SB2    X1                                            GRA      3158    A
GRA               NZ     B2,MFA35     IF LEN[AI] " 0                   GRA      3159    A
GRA                                                                    GRA      3160    A
GRA               SA2    A1-B1                                         GRA      3161    A
GRA               SA3    CIT+X2       AI = CIT(I)                      GRA      3162    A
GRA               SX7    X2+B1        I = I + 1                        GRA      3163    A
GRA               SA7    A2                                            GRA      3164    A
GRA               BX6    X3                                            GRA      3165    A
GRA               SA6    A1                                            GRA      3166    A
GRA               NZ     X3,MFA35     IF AI " 0                        GRA      3167    A
GRA                                                                    GRA      3168    A
GRA               EQ     MFA10                                         GRA      3169    A
GRA                                                                    GRA      3170    A
GRA       MFAC    BSS    0                                             GRA      3171    A
GRA               POS    AI.LENP+1                                     GRA      3172    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                VFD    1/1,*P/1                                                GRA        3173     A
GRA       CA.NOC2P EQU    CA.NOCCP+1                                              GRA        3174     A
GRA       CA.NOC2L EQU    CA.NOCCL-1                                              GRA        3175     A
GRA       MFAD     BFMW   CA,(FP,NOC2)                                            GRA        3176     A
GRA       MFA      SPACE  2,14                                                    GRA        3177     A
GRA      *         PARTIAL ASSIGNMENT, SET IH+CA IN A REGISTER                    GRA        3178     A
GRA                                                                               GRA        3179     A
GRA       MFA40    SX6    0                                                       GRA        3180     A
GRA                SA6    CAIH        CAIH = 0                                    GRA        3181     A
GRA                                                                               GRA        3182     A
GRA       MFA41    SX7    A5-B1                                                   GRA        3183     A
GRA                SB2    X5+B1                                                   GRA        3184     A
GRA                SA7    CIT                                                     GRA        3185     A
GRA                SB3    X5          J = CO[BI]                                  GRA        3186     A
GRA                SA4    B6+B2       C2W = RCT(J+1)                              GRA        3187     A
GRA                MX0    -IH.RFP                                                 GRA        3188     A
GRA                BX1    -X0*X4      CAIH1 = CAIH[C2W]                           GRA        3189     A
GRA                MX2    0                                                       GRA        3190     A
GRA                RJ     SDC         SEARCH FOR CAIH                             GRA        3191     A
GRA                NZ     B2,MFA45    IF REGN " 0   */ IN A REG                   GRA        3192     A
GRA                                                                               GRA        3193     A
GRA                SA1    MFAD                                                    GRA        3194     A
GRA                SA2    B6+B3       CAW = RCT(J)                                GRA        3195     A
GRA                SA4    NAB                                                     GRA        3196     A
GRA                BX3    X1*X2                                                   GRA        3197     A
GRA                ZR     X3,MFA47    IF ^( FP[CAW] ! NOCC[CAW] > 1 )             GRA        3198     A
GRA                ZR     X4,MFA47    IF NAB = 0                                  GRA        3199     A
GRA                                                                               GRA        3200     A
GRA                SA4    =XO.SYM                                                 CCGA100      10     A
GRA                SA5    A2+B1                                                   CCGA100      11     A
GRA                SX4    X4+B1                                                   CCGA100      12     A
GRA                LX5    -IH.IHP                                                 CCGA100      13     A
GRA                SX5    X5                                                      CCGA100      14     A
GRA                IX4    X4+X5                                                   CCGA100      15     A
GRA                LX5    1                                                       CCGA100      16     A
GRA                IX4    X4+X5                                                   CCGA100      17     A
GRA                SA4    X4          WORDB=SYM(3*IH+1)                           CCGA100      18     A
GRA                LX4    59-WB.LCMP                                              CCGA100      19     A
GRA                MI     X4,MFA47    IF LCM                                      CCGA100      20     A
GRA                                                                               CCGA100      21     A
GRA                RJ     EDC         ENTER CANIDATE                              GRA        3201     A
GRA                                                                               GRA        3202     A
GRA      *         SETUP *J* AS A *SLD*                                           GRA        3203     A
GRA                                                                               GRA        3204     A
GRA       MFA45    SA1    B6+B3                                                   GRA        3205     A
GRA                MX0    1                                                       GRA        3206     A
GRA                SA2    A1+2        CCW = RCT(J+2)                              GRA        3207     A
GRA                LX0    1+CA.RAP                                                GRA        3208     A
GRA                BX6    X0+X1       RA[RCT(J)] = 1                              GRA        3209     A
GRA                SA6    A1                                                      GRA        3210     A
GRA                SX7    2S6         MIT[CCW] = 2                                GRA        3211          I
          -FCC0646

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SX7    4S6          MIT[CCW] = 4  */ SPECIAL CASE SLD/SST/SA          FCC0646      21     A
GRA                SX5    A0                                                             GRA         3212     A
GRA                SX4    B2                                                             GRA         3213     A
GRA                ZR     X5,MFA46     IF REGNO = 0  */ NO BASE REG                      GRA         3214     A
GRA                                                                                     GRA         3215     A
GRA                BX7    X5+X7        REG1[CCW] = REGNO                                 GRA         3216     A
GRA                LX7    CC.REG1P                                                       GRA         3217     A
GRA                LX4    CC.REG2P     REG2[CCW] = REGN                                  GRA         3218     A
GRA                BX2    X4+X2                                                          GRA         3219     A
GRA                IX7    X2+X7                                                          GRA         3220          I
          -FCC1014
GRA                BX7    X2+X7                                                          FCC1014      103     A
GRA                SA7    A2                                                             GRA         3221     A
GRA                EQ     MFA47                                                          GRA         3222     A
GRA                                                                                     GRA         3223     A
GRA       MFA46    LX4    CC.REG1P                                                       GRA         3224     A
GRA                BX5    X4+X7        REG1[CCW] = REGN                                  GRA         3225     A
GRA                IX7    X5+X2                                                          GRA         3226          I
          -FCC1014
GRA                BX7    X5+X2                                                          FCC1014      104     A
GRA                SA7    A2                                                             GRA         3227     A
GRA                                                                                     GRA         3228     A
GRA       MFA47    SA4    CIT                                                            GRA         3229     A
GRA                SB5    B5-B1        L = L - 1                                         GRA         3230     A
GRA                SA5    X4           BI = BI - 1                                       GRA         3231     A
GRA                NZ     B5,MFA41     IF L " 0                                          GRA         3232     A
GRA                                                                                     GRA         3233     A
GRA                EQ     MFA10                                                          GRA         3234     A
GRA       MFA      SPACE  2,14                                                           GRA         3235     A
GRA       MFA48    SA1    NAB                                                            GRA         3236     A
GRA                SA2    N.HB                                                           GRA         3237     A
GRA                NZ     X1,MFA48A    IF NAB " 0    */ B-REGS AVAIL                     GRA         3238     A
GRA                SX7    X2-1                                                           GRA         3239     A
GRA                NZ     X7,MFA60     IF N.HB " 1   */ NOT OUTER LP WITH 1 INNER        GRA         3240     A
GRA                                                                                     GRA         3241     A
GRA      *         FORM SORT TABLE OF CONSTANTS WITH MORE THAN 1 OCCURANCE OR            GRA         3242     A
GRA      *         PREVIOUSLY ASSIGNED TO A REGISTER IN AN INNER LOOP.                   GRA         3243     A
GRA                                                                                     GRA         3244     A
GRA       MFA48A   SA2    CL                                                             GRA         3245     A
GRA                MX0    -CA.NOCCL                                                      GRA         3246     A
GRA                SX7    B0                                                             GRA         3247     A
GRA                LX0    CA.NOCCP                                                       GRA         3248     A
GRA                SB7    SCR+1                                                          GRA         3249     A
GRA                SA7    B7-B1        SI = SCR;  [SI] = 0                               GRA         3250     A
GRA                SX1    B0           L = 0                                             GRA         3251     A
GRA                SX2    X2           J = CL                                            GRA         3252     A
GRA                SB5    -36B                                                           GRA         3253     A
GRA                SA0    0            NPA = 0       */ N. PREVIOUS ASSIGNMENTS          GRA         3254     A
GRA                MX3    1                                                              GRA         3255     A
GRA                LX3    1+CA.RAP                                                       GRA         3256     A
GRA                                                                                     GRA         3257     A
GRA       MFA49    SA5    B6+X2        CAW = RCT(J)                                      GRA         3258     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                BX6    -X0*X5       N = NOCC[CAW]                          GRA        3259     A
GRA                IX7    X6+X2                                               GRA        3260     A
GRA                BX4    X3*X5                                               GRA        3261     A
GRA                LX4    58-CA.RAP                                           GRA        3262     A
GRA                BX7    X4+X7        STW = CST(RA[CAW],N,J)                 GRA        3263     A
GRA                ZR     X4,MFA50     IF ^RA[CAW]                            GRA        3264     A
GRA                SA0    A0+B1        NPA = NPA + 1                          GRA        3265     A
GRA                EQ     MFA51                                               GRA        3266     A
GRA                                                                           GRA        3267     A
GRA        MFA50   LX6    -CA.NOCCP                                           GRA        3268     A
GRA                SB2    X6                                                  GRA        3269     A
GRA                LE     B2,B1,MFA52 IF N @ 1                               GRA        3270     A
GRA                                                                           GRA        3271     A
GRA        MFA51   SX1    X1+1         L = L + 1                              GRA        3272     A
GRA                SA7    A7+B1        SI = SI + 1;  [SI] = CST(N,J)          GRA        3273     A
GRA                SB2    X1+B5                                               GRA        3274     A
GRA                PL     B2,MFA53     IF L = 36B                             GRA        3275     A
GRA                                                                           GRA        3276     A
GRA        MFA52   SA5    A5+2                                                GRA        3277     A
GRA                LX5    -CC.INCP                                            GRA        3278     A
GRA                SX2    X5           J = INC[RCT(J+2)]                      GRA        3279     A
GRA                NZ     X2,MFA49     IF J " 0      */ NOT END OF CON LIST   GRA        3280     A
GRA                                                                           GRA        3281     A
GRA        MFA53   ZR     X1,MFA60     IF L = 0      */ NO CANIDATES        GRA        3282     A
GRA                MX7    0                                                   GRA        3283     A
GRA                SA7    A7+B1        SI = SI + 1;  [SI] = 0                 GRA        3284     A
GRA                                                                           GRA        3285     A
GRA                CALL   SHL          SORT( SCR )                            GRA        3286          I
           -CCG1
GRA                CALL   SHL=         SORT( SCR )                            CCG1        114          I
           -CCGNC
GRA                CALL   SHL#         SORT( SCR )                            CCGNC       656     A
GRA                SA1    O.RND                                               GRA        3287     A
GRA                SA5    NAB                                                 GRA        3288     A
GRA                SB2    A0+B1                                               GRA        3289     A
GRA                SB3    B3-B2        SI = SI - (NPA+1)  */ LWA OF ^ASGNED CONS GRA      3290     A
GRA                SB7    X1+                                                 GRA        3291     A
GRA                NZ     X5,MFA54     IF NAB " 0                             GRA        3292     A
GRA                EQ     B2,B1,MFA60 IF NPA = 0                             GRA        3293     A
GRA                EQ     MFA55                                               GRA        3294     A
GRA                                                                           GRA        3295     A
GRA        *       ASSIGN CONSTANTS TO THE B-REGS                             GRA        3296     A
GRA                                                                           GRA        3297     A
GRA        MFA54   SA5    B3           J = CO[SI]                             GRA        3298     A
GRA                ZR     X5,MFA55     IF J = 0      */ END OF SORT TABLE     GRA        3299     A
GRA                SB3    B3-B1        SI = SI - 1                            GRA        3300     A
GRA                SX1    B1           INVF = 1                               GRA        3301     A
GRA                SB4    X5                                                  GRA        3302     A
GRA                RJ     AIR          ASSIGN AN INDEX REGISTER               GRA        3303     A
GRA                SA1    NAB                                                 GRA        3304     A
GRA                NZ     X1,MFA54     IF NAB " 0    */ MORE B-REGS LEFT      GRA        3305     A
GRA                                                                           GRA        3306     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA      *         PROPAGATE CONSTANT ASSIGNMENTS TO PARTIALLY MODIFIED INSTS          GRA      3307     A
GRA                                                                                    GRA      3308     A
GRA       MFA55    SA2    PMF                                                          GRA      3309     A
GRA                ZR     X2,MFA60    IF PMF = 0    */ NO PARTIAL MODS                 GRA      3310     A
GRA                MX0    -IH.CAL                                                      GRA      3311     A
GRA                SA3    B3+B1       SI = SI + 1   */ IN *A3*                         GRA      3312     A
GRA                LX0    IH.CAP                                                       GRA      3313     A
GRA                SB4    X3+B1       J = CO[SI]                                       GRA      3314     A
GRA                                                                                    GRA      3315     A
GRA       MFA56    SA1    B6+B4       C2W = RCT(J+1)                                   GRA      3316     A
GRA                SA2    A1+B1                                                        GRA      3317     A
GRA                MX7    -CC.REG1L                                                    GRA      3318     A
GRA                LX2    -CC.REG1P                                                    GRA      3319     A
GRA                BX2    -X7*X2      REGN = REG1[RCT(J+2)]                            GRA      3320     A
GRA                SA5    B7+B1       BI = O.RND + 1                                   GRA      3321     A
GRA                LX2    CC.REG2P                                                     GRA      3322     A
GRA                                                                                    GRA      3323     A
GRA       MFA57    PL     X5,MFA59    IF [BI] > 0   */ NOT A PARTIAL MOD               GRA      3324     A
GRA                BX5    -X5                                                          GRA      3325     A
GRA                SB2    X5+B1       I = -CO[BI]                                      GRA      3326     A
GRA                SA4    B6+B2                                                        GRA      3327     A
GRA                SX7    1S6         IT = 2                                           GRA      3328     A
GRA                BX5    -X0*X4      C = CA[RCT(I+1)]                                 GRA      3329     A
GRA                IX6    X5-X1                                                        GRA      3330     A
GRA                ZR     X6,MFA58    IF C = C2W                                       GRA      3331     A
GRA                IX4    X5+X1                                                        GRA      3332     A
GRA                BX6    X0+X4                                                        GRA      3333     A
GRA                NZ     X6,MFA59    IF -C " C2W                                      GRA      3334     A
GRA                LX7    1           IT = 3        */ 2S6                            GRA      3335     A
GRA                                                                                    GRA      3336     A
GRA      *         CHANGE INSTRUCTION                                                 GRA      3337     A
GRA                                                                                    GRA      3338     A
GRA       MFA58    SA4    A4+B1       CCW = RCT(I+2)                                   GRA      3339     A
GRA                BX6    X2+X7       REG2[CCW] = REGN                                 GRA      3340     A
GRA                IX7    X6+X4       MIT[CCW] = IT                                    GRA      3341     A
GRA                SA6    A4                                                           GRA      3342     A
GRA                SA7    A4                                                           GRA      3343     A
GRA                                                                                    GRA      3344     A
GRA       MFA59    SA5    A5+B1       BI = BI + 1                                      GRA      3345     A
GRA                NZ     X5,MFA57    IF [BI] " 0   */ NOT END OF ADDRESS TABLE        GRA      3346     A
GRA                                                                                    GRA      3347     A
GRA                SA3    A3+B1       SI = SI + 1                                      GRA      3348     A
GRA                SB4    X3+B1       J = CO[SI]                                       GRA      3349     A
GRA                NZ     X3,MFA56    IF J " 0                                         GRA      3350     A
GRA       MFA      SPACE  2,14                                                         GRA      3351     A
GRA      *         SETUP LOOP TEST REPLACEMENT CODE INFO                              GRA      3352     A
GRA                                                                                    GRA      3353     A
GRA       MFA60    SA1    TRD                                                          GRA      3354     A
GRA                SX6    0           L.RND = 0                                        GRA      3355     A
GRA                SA6    L.RND                                                        GRA      3356     A
GRA                ZR     X1,MFA      IF TRD = 0    */ NO TEST REPLACEMENT             GRA      3357     A
GRA                                                                                    GRA      3358     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SA5    LUL                                                    GRA        3359    A
GRA                UX6    B2,X5                                                  GRA        3360    A
GRA                NZ     B2,MFA61    IF TYPE[LUL] " 0  */ NOT A CON *UL*        GRA        3361    A
GRA                                                                              GRA        3362    A
GRA      *         ENTER CONSTANT UPPER LIMIT IN *RCT*                            GRA        3363    A
GRA                                                                              GRA        3364    A
GRA                ALLOC  RCT,3       ALLOC( RCT , 3 )                           GRA        3365    A
GRA                MX0    -IH.CAL                                                GRA        3366    A
GRA                SX6    B1                                                     GRA        3367    A
GRA                BX7    -X0*X5                                                 GRA        3368    A
GRA                LX6    CA.CONP                                                GRA        3369    A
GRA                SA6    X2+B6       RCT(OL) = CAW(0,CON,0)                     GRA        3370    A
GRA                LX7    IH.CAP                                                 GRA        3371    A
GRA                SA7    A6+B1       RCT(OL+1) = IHW(0,0,CA[LUL],0)             GRA        3372    A
GRA                MX6    0                                                      GRA        3373    A
GRA                SA6    A7+B1       RCT(OL+2) = 0                              GRA        3374    A
GRA                SA1    TRD                                                    GRA        3375    A
GRA                SA4    X2+B1                                                  GRA        3376    A
GRA                SX6    X1-2                                                   GRA        3377    A
GRA                NZ     X6,MFA60B   IF TRD " 2                                 GRA        3378    A
GRA                                                                              GRA        3379    A
GRA      *         I, N IN REGS, CHECK FOR PREVIOUS ASSIGNMENT OF *UL* IN THIS LP GRA        3380    A
GRA                                                                              GRA        3381    A
GRA                SB5    X2+4                                                   GRA        3382    A
GRA                SB3    3                                                      GRA        3383    A
GRA       MFA60A   BX6    X4-X7                                                  GRA        3384    A
GRA                SA4    A4+B3                                                  GRA        3385    A
GRA                NZ     X6,MFA60A   IF RCT(I-2) " RCT(OL+1)                    GRA        3386    A
GRA                SB4    A4-B5                                                  GRA        3387    A
GRA                EQ     B4,B6,MFA60B      IF I = OL                            GRA        3388    A
GRA                SA3    A4-4        CAW = RCT(I)                               GRA        3389    A
GRA                LX3    59-CA.RAP                                              GRA        3390    A
GRA                PL     X3,MFA60A   IF ^RA[RCT(I)-3)]                          GRA        3391    A
GRA                LX3    CA.RAP-CA.CONP                                         GRA        3392    A
GRA                PL     X3,MFA60A   IF ^CON[CAW]                               GRA        3393    A
GRA                SA1    LUL+1                                                  GRA        3394    A
GRA                SA6    RAT+X1      RAT(LUL(2)) = 0                            GRA        3395    A
GRA                SA4    A3+2        CCW = RCT(I+2)                             GRA        3396    A
GRA                LX4    -CC.REG1P                                              GRA        3397    A
GRA                MX0    -CC.REG1L                                              GRA        3398    A
GRA                BX6    -X0*X4      LUL(2) = REG1[CCW]                         GRA        3399    A
GRA                SA6    A1                                                     GRA        3400    A
GRA                SB6    B4          OL = I                                     GRA        3401    A
GRA                                                                              GRA        3402    A
GRA       MFA60B   SA1    LUL+1                                                  GRA        3403    A
GRA                SX7    B6                                                     GRA        3404    A
GRA                MX0    1                                                      GRA        3405    A
GRA                LX0    1+RA.INVP                                              GRA        3406    A
GRA                BX7    X0+X7       RAT(LUL(2)) = RATW(1,0,0,OL)               GRA        3407    A
GRA                SA7    RAT+X1                                                 GRA        3408    A
GRA                                                                              GRA        3409    A
GRA       MFA61    SA5    TRD                                                    GRA        3410    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA              SB2    X5-2                                                GRA        3411    A
GRA              ZR     B2,MFA      IF TRD = 2    */ I,N CASE               GRA        3412    A
GRA              PL     B2,MFA63    IF TRD " 1                              GRA        3413    A
GRA                                                                         GRA        3414    A
GRA      *       I-N CASE, SETUP TO BIAS REG THAT *CV* IS IN                GRA        3415    A
GRA                                                                         GRA        3416    A
GRA       MFA62  SA1    LUL+1                                               GRA        3417    A
GRA              SA2    RAT+X1                                              GRA        3418    A
GRA              SA3    LCV+1                                               GRA        3419    A
GRA              SA4    RAT+X3                                              GRA        3420    A
GRA              SX2    X2                                                  GRA        3421    A
GRA              LX2    RA.R2P      R2[RAT(LCV(2))] = R1[RAT(LUL(2))]       GRA        3422    A
GRA              BX6    X2+X4                                               GRA        3423    A
GRA              SA6    A4                                                  GRA        3424    A
GRA              MX7    0                                                   GRA        3425    A
GRA              SA7    A2          RAT(LUL(2)) = 0                         GRA        3426    A
GRA              SA7    A1          LUL(2) = 0                              GRA        3427    A
GRA              EQ     MFA                                                 GRA        3428    A
GRA                                                                         GRA        3429    A
GRA      *       PROCESS A+I , A+N CASE , CV REG IS ALREADY BIASED          GRA        3430    A
GRA                                                                         GRA        3431    A
GRA       MFA63  NE     B2,B1,MFA64 IF TRD " 3                              GRA        3432    A
GRA              SA1    LUL+1                                               GRA        3433    A
GRA              SA2    RAT+X1                                              GRA        3434    A
GRA              SA3    LCV+1                                               GRA        3435    A
GRA              SA4    RAT+X3                                              GRA        3436    A
GRA              MX0    -RA.R3L                                             GRA        3437    A
GRA              SX7    B1                                                  GRA        3438    A
GRA              LX0    RA.R3P                                              GRA        3439    A
GRA              BX5    -X0*X4      J = R3[RAT(LCV(2))]                     GRA        3440    A
GRA              LX7    RA.INVP                                             GRA        3441    A
GRA              BX2    X7+X2       INV[RAT(LUL(2))] = 1                    GRA        3442    A
GRA              BX6    X5+X2       R3[RAT(LUL(2))] = J                     GRA        3443    A
GRA              SA6    A2                                                  GRA        3444    A
GRA              EQ     MFA                                                 GRA        3445    A
GRA                                                                         GRA        3446    A
GRA      *       TRD = 4, I,F(I) IN REGS, SEARCH FOR F(I)                   GRA        3447    A
GRA                                                                         GRA        3448    A
GRA       MFA64  SA1    O.RCT                                               GRA        3449    A
GRA              SA2    O.TET                                               GRA        3450    A
GRA              SA3    O.IIT                                               GRA        3451    A
GRA              SA4    =XIT.                                               GRA        3452    A
GRA              SA5    LCV                                                 GRA        3453    A
GRA              SB7    7           I = 7         */ REGNO                  GRA        3454    A
GRA              SB6    X1+B1                                               GRA        3455    A
GRA              SB5    X2                                                  GRA        3456    A
GRA              SB4    X3+B1                                               GRA        3457    A
GRA              BX4    -X4                                                 GRA        3458    A
GRA              SB3    X4                                                  GRA        3459    A
GRA              UX5    X5                                                  GRA        3460    A
GRA              AX5    PS.UDIP                                             GRA        3461    A
GRA              BX5    -X5                                                 GRA        3462    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SB2    X5          U = UDI[LCV]                                GRA        3463     A
GRA                                                                               GRA        3464     A
GRA       MFA65    SA5    RAT+B7      R = R1[RAT(I)]                              GRA        3465     A
GRA                ZR     X5,MFA66    IF R = 0                                    GRA        3466     A
GRA                SA4    B6+X5       C2W = RCT(R+1)                              GRA        3467     A
GRA                SX6    X4+B3                                                   GRA        3468     A
GRA                NZ     X6,MFA66    IF IH[C2W] " IT.                            GRA        3469     A
GRA                SA3    A4-B1                                                   GRA        3470     A
GRA                LX4    -IH.CAP     C = CA[C2W]                                 GRA        3471     A
GRA                LX3    59-CA.KDP                                               GRA        3472     A
GRA                SA2    B5+X4                                                   GRA        3473     A
GRA                LX2    -T.ITIP     ITI = ITI[TET(C)]                           GRA        3474     A
GRA                MI     X3,MFA66    IF KD[RCT(R)]                               GRA        3475     A
GRA                SX2    X2                                                      GRA        3476     A
GRA                ZR     X2,MFA66    IF ITI = 0                                  GRA        3477     A
GRA                SA1    B4+X2       IAW = IIT(ITI+1)                            GRA        3478     A
GRA                LX1    -IA.UDIP                                                GRA        3479     A
GRA                SX6    X1+B2                                                   GRA        3480     A
GRA                ZR     X6,MFA67    IF UDI[IAW] = U                             GRA        3481     A
GRA                                                                               GRA        3482     A
GRA       MFA66    SB7    B7-B1       I = I - 1                                   GRA        3483     A
GRA                NZ     B7,MFA65    IF I " 0                                    GRA        3484     A
GRA                                                                               GRA        3485     A
GRA      *         F(I) NOT FOUND, CHANGE TO TRD = 1                              GRA        3486     A
GRA                                                                               GRA        3487     A
GRA                SX6    B1                                                      GRA        3488     A
GRA                SA6    TRD         TRD = 1                                     GRA        3489     A
GRA                EQ     MFA62                                                   GRA        3490     A
GRA                                                                               GRA        3491     A
GRA      *         POSSIBLE F(I) FOUND, CHECK INCREMENT VALUE. DISALLOW VARIABLE  GRA        3492     A
GRA      *         INCREMENTS SINCE THEY MAY BE NEGATIVE.                         GRA        3493     A
GRA                                                                               GRA        3494     A
GRA       MFA67    SA5    A4+B1       CCW = RCT(R+2)                              GRA        3495          I
         -FCC2135
GRA                LX5    -CC.INCP                                                GRA        3496          I
         -FCC2135
GRA                SX6    X5-1        IP = INC[CCW]                               GRA        3497          I
         -FCC2135
GRA                SA1    B6+X6       CAWI = RCT(IP)                              GRA        3498          I
         -FCC2135
GRA                SX7    X1                                                      GRA        3499          I
         -FCC2135
GRA                LX1    59-CA.CONP                                              GRA        3500          I
         -FCC2135
GRA                MI     X1,MFA68    IF CON[CAWI]  */ CON INCR                   GRA        3501          I
         -FCC2135
GRA                ZR     X7,MFA66    IF UDI[CAWI] = 0  */ COMPILER TEMP          GRA        3502          I
         -FCC2135
GRA                EQ     MFA66       MORE CODE TO ALLOW PROGRAMMER VAR           GRA        3503          I
         -FCC2135
GRA       MFA67    SA3    A4+B1       CCW=RCT(R+2)                       001250   FCC2135       4     A
GRA                LX3    -CC.INCP                                       001260   FCC2135       5     A

         0        1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SX6    X3-1        IP=INC[CCW]                          001270   FCC2135      6    A
GRA                SA1    B6+X6       CAWI=RCT(IP)                         001280   FCC2135      7    A
GRA                LX1    59-CA.CONP                                       001290   FCC2135      8    A
GRA                PL     X1,MFA66    IF CON[CAWI]=0 */VAR INCR.           001300   FCC2135      9    A
GRA                                                                        001310   FCC2135     10    A
GRA       *   IF F(I) INDEXES BLANK COMMON, A FORMAL PARAMETER OR LCM AND ITS  001320   FCC2135     11    A
GRA       *   INCREMENT IS SUFFICIENTLY LARGE USING IT AS AN ALTERNATE LOOP COUNT 001330   FCC2135     12    A
GRA       *   COULD RESULT IN A B REGISTER OVERFLOW.  FIRST CHECK IF SIZE OF   001340   FCC2135     13    A
GRA       *   INCREMENT IS SUFFICIENTLY LARGE (>100B).                         001350   FCC2135     14    A
GRA                                                                        001360   FCC2135     15    A
GRA                SA1    A1+B1       IHWI=CAWI+1                          001370   FCC2135     16    A
GRA                LX1    -IH.CAP-6                                        001380   FCC2135     17    A
GRA                SX1    X1                                               001390   FCC2135     18    A
GRA                ZR     X1,MFA68    IF CA[IHWI]<100B                     001400   FCC2135     19    A
GRA                                                                        001410   FCC2135     20    A
GRA       *   NOW CHECK IF F(I) INDEXES BLANK COMMON,A FORMAL PARAMETER OR LCM.  001420   FCC2135     21    A
GRA                                                                        001430   FCC2135     22    A
GRA                SX5    X5          LOOK THRU IH WORDS OF RCT            001440   FCC2135     23    A
GRA                MX6    -IH.RFL     BACKWARDS LOOKING FOR THE            001450   FCC2135     24    A
GRA                SX0    B2          F(I) RCT INDEX IN THE                001460   FCC2135     25    A
GRA                LX5    IH.RFP      RF FIELD.                            001470   FCC2135     26    A
GRA                SA1    L.RCT                                            001480   FCC2135     27    A
GRA                SA4    =X0.SYM                                          001490   FCC2135     28    A
GRA                LX6    IH.RFP                                           001500   FCC2135     29    A
GRA                SA1    B6+X1                                            001510   FCC2135     30    A
GRA                SA0    X4+B1                                            001520   FCC2135     31    A
GRA                                                                        001530   FCC2135     32    A
GRA        MFA67A  SA1    A1-3                                             001540   FCC2135     33    A
GRA                ZR     X1,MFA68    IF END OF RCT                        001550   FCC2135     34    A
GRA                BX4    -X6*X1                                           001560   FCC2135     35    A
GRA                BX4    X4-X5                                            001570   FCC2135     36    A
GRA                NZ     X4,MFA67A   IF F(I) NE RF                        001580   FCC2135     37    A
GRA                LX1    -IH.IHP     IH=IH[RCT(I+1)]                      001590   FCC2135     38    A
GRA                SB2    X1                                               001600   FCC2135     39    A
GRA                ZR     B2,MFA67B   IF IH = 0 THEN UNSAFE
>>>>     CCGA036 24                                                                CCGA036     18    A
GRA                LX1    1                                                001610   FCC2135     40    A
GRA                SB2    B2+X1                                            001620   FCC2135     41    A
GRA                SA4    A0+B2       WORDB=SYM(3*IH+1)                    001630   FCC2135     42    A
GRA                LX4    59-WB.LCMP                                       001640   FCC2135     43    A
GRA                MI     X4,MFA67B   IF LCM                               001650   FCC2135     44    A
GRA                LX4    WB.LCMP-WB.FPP                                   001660   FCC2135     45    A
GRA                MI     X4,MFA67B   IF A FORMAL PARAMETER                001670   FCC2135     46    A
GRA                LX4    WB.FPP-WB.COMP                                   001680   FCC2135     47    A
GRA                PL     X4,MFA67A   IF NOT COMMON                        001690   FCC2135     48    A
GRA                SA4    A4+B1       WORDC=WORDB+1                        001700   FCC2135     49    A
GRA                MX7    -WC.RBL                                          001710   FCC2135     50    A
GRA                LX4    -WC.RBP                                          001720   FCC2135     51    A
GRA                BX7    -X7*X4      RBN=RB[WORDC]                        001730   FCC2135     52    A
GRA        .FTN    IFEQ   HC.ID,2,2                                        001740   FCC2135     53    A
GRA                SA4    =XBLNKCOM                                        001750   FCC2135     54    A
GRA                BX7    X4-X7                                            001760   FCC2135     55    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                NZ     X7,MFA67A   IF RBN=0 */INDEXES BLANK COMMON          001770   FCC2135      56     A
GRA                                                                             001780   FCC2135      57     A
GRA       *   F(I) IS NOT SAFE LOOK FOR ANOTHER F(I).                           001790   FCC2135      58     A
GRA                                                                             001800   FCC2135      59     A
GRA        MFA67B  SB2    X0          RESTORE B2                               001810   FCC2135      60     A
GRA                EQ     MFA66                                                001820   FCC2135      61     A
GRA                                                                                      GRA        3504     A
GRA       *        ADJUST TRD IF NEGATIVE F(I), SAVE *IIT* INDEX                           GRA        3505     A
GRA                                                                                      GRA        3506     A
GRA        MFA68   LX5    59-CC.IMP+CC.INCP                                               GRA        3507          I
          -FCC2135
GRA        MFA68   LX3    59-CC.IMP+CC.INCP                                       001840   FCC2135      62     A
GRA                MX7    1                                                                GRA        3508     A
GRA                SX6    4                                                                GRA        3509     A
GRA                BX7    X7*X5                                                            GRA        3510          I
          -FCC2135
GRA                BX7    X7*X3                                                   001860   FCC2135      63     A
GRA                BX6    X6+X7       TRD = SHIFT(IM[CCW],59) ! 4                          GRA        3511     A
GRA                SA6    TRD                                                              GRA        3512     A
GRA                SA1    RAT         ULO = R1[RAT(0)]  */ UL ORD                          GRA        3513     A
GRA                SA3    LCV+1                                                            GRA        3514     A
GRA                SA4    RAT+X3      CVO = R1[RAT(LCV(2))]                                GRA        3515     A
GRA                BX6    X3                                                               GRA        3516     A
GRA                SX7    B7                                                               GRA        3517     A
GRA                SA6    LUL+1       J = LUL(2) = LCV(2)  */ UL IN CV REG                 GRA        3518     A
GRA                SA7    A3          LCV(2) = I    */ CV IN F(I) REG                      GRA        3519     A
GRA                SX2    X2+B1                                                            GRA        3520     A
GRA                LX2    RA.R2P      R2[RAT(J)] = ITI + 1  */ FORMULA INDEX               GRA        3521     A
GRA                BX7    X1+X2       R1[RAT(J)] = ULO                                     GRA        3522     A
GRA                SA7    RAT+X3                                                           GRA        3523     A
GRA                MX6    0                                                                GRA        3524     A
GRA                SA6    A1          RAT(0) = 0                                           GRA        3525     A
GRA                SB6    B6-B1                                                            GRA        3526     A
GRA                MX7    1                                                                GRA        3527     A
GRA                SA2    SEEA                                                             GRA        3528     A
GRA                SA3    B6+X4                                                            GRA        3529     A
GRA                BX6    X2+X3       KD[RCT(CVO)] = 1                                     GRA        3530     A
GRA                SA6    A3                                                               GRA        3531     A
GRA                EQ     MFA                                                              GRA        3532     A
GRA         PSC    TITLE  PSC - PROCESS SIMPLE CLASS ( 1 IH )                              GRA        3533     A
GRA       **       PSC - PROCESS SIMPLE CLASS ( 1 IH )                                     GRA        3534     A
GRA       *        FIND A BASE MEMBER ( *CA* IS IN CENTER OF CLASS ).                      GRA        3535     A
GRA       *        REDUCE THE BASE MEMBER TO A SHORT INSTRUCTION AND CHANGE                GRA        3536     A
GRA       *        THE OTHER MEMBERS OF THE CLASS TO THE FORM *REG+CON*.                   GRA        3537     A
GRA       *                                                                                GRA        3538     A
GRA       *        ENTRY  (B5) = L , CLASS LENGTH                                          GRA        3539     A
GRA       *               (A5) = BI , LWA OF CLASS                                         GRA        3540     A
GRA       *               (A0) = REGNO , REGISTER NUMBER OF *RF*                           GRA        3541     A
GRA       *                                                                                GRA        3542     A
GRA       *        EXIT   (CAIH) = CAIH OF BASE MEMBER                                     GRA        3543     A
GRA                                                                                      GRA        3544     A
GRA         PSC    ROUTINE                                                                 GRA        3545     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SB4    B5-B1                                                GRA        3546     A
GRA                SB6    B6+B1                                                GRA        3547     A
GRA                LE     B4,B1,PSC5  IF L @ 2      */ ONLY 2 MEMBERS          GRA        3548     A
GRA                                                                            GRA        3549     A
GRA      *         L > 2 ,  FIND A BASE MEMBER WHICH IS THE CENTER OF THE      GRA        3550     A
GRA      *         *CA*"S IN THE CLASS.                                        GRA        3551     A
GRA                                                                            GRA        3552     A
GRA                SA4    A5          I = BI                                   GRA        3553     A
GRA                MX6    0           AVE = 0                                  GRA        3554     A
GRA                SB4    B0          J = 0                                    GRA        3555     A
GRA                                                                            GRA        3556     A
GRA       PSC2     SA3    B6+X4       C2W = RCT(CO[I]+1)                       GRA        3557     A
GRA                SB4    B4+B1       J = J + 1                                GRA        3558     A
GRA                LX3    -IH.CAP                                              GRA        3559     A
GRA                SA4    A4-B1       I = I - 1                                GRA        3560     A
GRA                SX7    X3                                                   GRA        3561     A
GRA                IX6    X6+X7       AVE = AVE + CA[C2W]                      GRA        3562     A
GRA                LT     B4,B5,PSC2  IF J < L                                 GRA        3563     A
GRA                SX7    B5                                                   GRA        3564     A
GRA                IX7    X6/X7,B4    AVE = AVE / L                            GRA        3565     A
GRA                                                                            GRA        3566     A
GRA      *         SEARCH CLASS TO FIND BASE MEMBER, AND MOVE IT TO LWA OF CLASS GRA      3567     A
GRA                                                                            GRA        3568     A
GRA                SB4    B0          J = 0                                    GRA        3569     A
GRA                SA4    A5          I = BI                                   GRA        3570     A
GRA                SB3    377777B     MIND = 377777B  */ MIN DIFFERENCE        GRA        3571     A
GRA                SX0    A5          IM = BI                                  GRA        3572     A
GRA                                                                            GRA        3573     A
GRA       PSC3     SA3    B6+X4       C2W = RCT(CO[I]+1)                       GRA        3574     A
GRA                LX3    -IH.CAP                                              GRA        3575     A
GRA                SX2    X3          C = CA[C2W]                              GRA        3576     A
GRA                IX6    X2-X7                                                GRA        3577     A
GRA                BX2    X6                                                   GRA        3578     A
GRA                AX6    59                                                   GRA        3579     A
GRA                BX2    X6-X2       DIFF = ABS( AVE - C )                    GRA        3580     A
GRA                SB2    X2                                                   GRA        3581     A
GRA                GE     B2,B3,PSC4  IF DIFF \ MIND                           GRA        3582     A
GRA                SX0    A4          IM = I                                   GRA        3583     A
GRA                SB3    B2          MIND = DIFF                              GRA        3584     A
GRA       PSC4     SB4    B4+B1       J = J + 1                                GRA        3585     A
GRA                SA4    A4-B1       I = I - 1                                GRA        3586     A
GRA                LT     B4,B5,PSC3  IF J < L                                 GRA        3587     A
GRA                                                                            GRA        3588     A
GRA                SA4    X0                                                   GRA        3589     A
GRA                LX6    X5                                                   GRA        3590     A
GRA                LX7    X4          SWAP( [IM] , [BI] )                      GRA        3591     A
GRA                SA6    A4                                                   GRA        3592     A
GRA                SA7    A5                                                   GRA        3593     A
GRA                                                                            GRA        3594     A
GRA      *         CHANGE CLASS BASE TO A SHORT REF                            GRA        3595     A
GRA                                                                            GRA        3596     A
GRA       PSC5     SA5    A5                                                   GRA        3597     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SB6    B6-B1                                                GRA        3598    A
GRA                RJ     AFA         ASSIGN FULL ADDRESS TO BASE              GRA        3599    A
GRA                                                                             GRA        3600    A
GRA      *         CHANGE REST OF CLASS TO LONG REFS OF FORM REG+CON           GRA        3601    A
GRA                                                                             GRA        3602    A
GRA                SA1    A6-B1       C2W = CCW - 1                            GRA        3603    A
GRA                MX0    -IH.RFP                                              GRA        3604    A
GRA                BX7    -X0*X1                                               GRA        3605    A
GRA                LX1    -IH.CAP                                              GRA        3606    A
GRA                SB4    X1          BCA = CA[C2W] */ CA THAT IS IN BASE REG  GRA        3607    A
GRA                SA7    CAIH        CAIH = CAIH[C2W]                         GRA        3608    A
GRA                SB4    -B4                                                  GRA        3609    A
GRA                EQ     B5,B1,PSC   IF L = 1                                 GRA        3610    A
GRA                                                                             GRA        3611    A
GRA       PSC6     SA5    A5-B1       BI = BI - 1                              GRA        3612    A
GRA                MX0    -IH.CAL                                              GRA        3613    A
GRA                SB2    X5+B1       J = CO[BI]                               GRA        3614    A
GRA                SA4    B6+B2       C2W = RCT(J+1)                           GRA        3615    A
GRA                BX6    -X5         [BI] = -[BI]  */ SET FLAG FOR LATER      GRA        3616    A
GRA                SA6    A5                                                   GRA        3617    A
GRA                LX4    -IH.CAP                                              GRA        3618    A
GRA                SX3    X4+B4       C = CA[C2W] - BCA  */ CHANGE CA          GRA        3619    A
GRA                BX4    X0*X4                                                GRA        3620    A
GRA                SA2    A4-B1       CAW = RCT(J)                             GRA        3621    A
GRA                BX1    -X0*X3                                               GRA        3622    A
GRA                BX6    X4+X1       CA[C2W] = C                              GRA        3623    A
GRA                LX6    IH.CAP                                               GRA        3624    A
GRA                MX1    1                                                    GRA        3625    A
GRA                SA6    A4                                                   GRA        3626    A
GRA                LX1    1+CA.RAP                                             GRA        3627    A
GRA                BX7    X1+X2       RA[RCT(J)] = 1                           GRA        3628    A
GRA                SA7    A2                                                   GRA        3629    A
GRA                SA1    A4+B1       CCW = RCT(J+2)                           GRA        3630    A
GRA                SX7    A0+1S6                                               GRA        3631    A
GRA                LX7    CC.REG1P    REG1[CCW] = REGNO                        GRA        3632    A
GRA                BX6    X7+X1       MIT[CCW] = 1  */ PLD, PST                GRA        3633    A
GRA                SA6    A1                                                   GRA        3634    A
GRA                RJ     ECC         ENTER CON CANIDATE                       GRA        3635    A
GRA                SB5    B5-B1       L = L - 1                                GRA        3636    A
GRA                GT     B5,B1,PSC6  IF L > 1                                 GRA        3637    A
GRA                                                                             GRA        3638    A
GRA                SX6    B1                                                   GRA        3639    A
GRA                SA6    PMF         PMF = 1       */ INDICATE PLD"S EXIST    GRA        3640    A
GRA                EQ     PSC                                                  GRA        3641    A
GRA       AAD      TITLE  AAD - ASSIGN ADDRESS DIFFERENCE TO A REGISTER        GRA        3642    A
GRA      **        AAD - ASSIGN ADDRESS DIFFERENCE TO A REGISTER               GRA        3643    A
GRA      *                                                                      GRA        3644    A
GRA      *         ENTRY  (A0) = REGNO OF BASE REGISTER                        GRA        3645    A
GRA      *                (B5) = J , ORDINAL OF CANIDATE TO BE DIFFERENCED     GRA        3646    A
GRA      *                (CAIH) = CA,IH IN BASE REGISTER                      GRA        3647    A
GRA                                                                             GRA        3648    A
GRA       AAD      ROUTINE                                                      GRA        3649    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA              SA1    CAIH        CAIH1 = CAIH                                    GRA        3650    A
GRA              SB2    B5+B1                                                       GRA        3651    A
GRA              SA2    B6+B2       CAIH2 = RCT(J+1)                                GRA        3652    A
GRA              RJ     SDC         SEARCH FOR CAIH - RCT(J+1)                      GRA        3653    A
GRA              SB3    3S6         IT = 3        */ SDL, SDS                       GRA        3654    A
GRA              NZ     B2,AAD1     IF REGN " 0   */ DIFF IN A REGISTER             GRA        3655    A
GRA              SA2    CAIH                                                        GRA        3656    A
GRA              SB2    B5+B1                                                       GRA        3657    A
GRA              SA1    B6+B2                                                       GRA        3658    A
GRA              RJ     SDC         SEARCH FOR RCT(J+1) - CAIH                      GRA        3659    A
GRA              SB3    2S6         IT = 2        */ SLD, SST                       GRA        3660    A
GRA              NZ     B2,AAD1     IF REGN " 0                                     GRA        3661    A
GRA                                                                                GRA        3662    A
GRA              RJ     EDC         ENTER DIFFERENCE CANIDATE                       GRA        3663    A
GRA                                                                                GRA        3664    A
GRA      *       SETUP *J* AS A SLD OR SDL                                          GRA        3665    A
GRA                                                                                GRA        3666    A
GRA       AAD1   SA1    B6+B5                                                       GRA        3667    A
GRA              MX0    1                                                           GRA        3668    A
GRA              SA2    A1+2        CCW = RCT(J+2)                                  GRA        3669    A
GRA              LX0    1+CA.RAP                                                    GRA        3670    A
GRA              SX3    A0+B3       REG1[CCW] = REGNO;  MIT[CCW] = IT               GRA        3671    A
GRA              LX3    CC.REG1P                                                    GRA        3672    A
GRA              SX4    B2          REG2[CCW] = REGN                                GRA        3673    A
GRA              LX4    CC.REG2P                                                    GRA        3674    A
GRA              BX6    X0+X1       RA[RCT(J)] = 1                                  GRA        3675    A
GRA              IX5    X3+X4                                                       GRA        3676    A
GRA              SA6    A1                                                          GRA        3677    A
GRA              BX7    X5+X2                                                       GRA        3678    A
GRA              SA7    A2                                                          GRA        3679    A
GRA              EQ     AAD                                                         GRA        3680    A
GRA       ECD    SPACE  3,14        ECD                                             GRA        3681    A
GRA      **      EDC - ENTER DIFFERENCE CANIDATE IN *RCT* AND ASSIGN A REG          GRA        3682    A
GRA                                                                                GRA        3683    A
GRA       EDC    ROUTINE                                                            GRA        3684    A
GRA              SA1    L.RCT                                                       GRA        3685    A
GRA              SA4    EDCA                                                        GRA        3686    A
GRA              SB2    X1+B1       OL = L.RCT                                      GRA        3687    A
GRA              SA6    B6+B2       RCT(OL+1) = KEY                                 GRA        3688    A
GRA              SA7    A6+B1       RCT(OL+2) = KEY2                                GRA        3689    A
GRA              BX6    X4                                                          GRA        3690    A
GRA              SA6    A6-B1       RCT(OL) = CAW(0,AC,PL,0)                        GRA        3691    A
GRA              SX7    X1+3        L.RCT = L.RCT + 3                               GRA        3692    A
GRA              SA7    A1                                                          GRA        3693    A
GRA              SB4    X1                                                          GRA        3694    A
GRA              SX1    B1          INVF = 1                                        GRA        3695    A
GRA              RJ     AIR         ASSIGN A B-REG                                  GRA        3696    A
GRA              SB2    -B2                                                         GRA        3697    A
GRA              EQ     EDC                                                         GRA        3698    A
GRA                                                                                GRA        3699    A
GRA       EDCA   BFMW   CA,(AC,PL)                                                 GRA        3700    A
GRA       SDC    SPACE  3,20        SDC                                             GRA        3701    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA      **        SDC - SEARCH FOR DIFFERENCE CANIDATE                          GRA      3702     A
GRA      *                                                                       GRA      3703     A
GRA      *         ENTRY  (X1) = CAIH1                                           GRA      3704     A
GRA      *                (X2) = CAIH2                                           GRA      3705     A
GRA      *                                                                       GRA      3706     A
GRA      *         EXIT   (X6) = KEY = IHW(0,0, CA1-CA2 , IH1 )                  GRA      3707     A
GRA      *                (X7) = KEY2 = CCW(0,IH2,0,0,0)                         GRA      3708     A
GRA      *                (B2) = REGN , REGNO THAT KEY,KEY2 IS IN                GRA      3709     A
GRA                                                                              GRA      3710     A
GRA       SDC      ROUTINE                                                       GRA      3711     A
GRA                SX7    X2                                                     GRA      3712     A
GRA                LX7    CC.H2P      KEY2 = CCW(0,IH2,0)                        GRA      3713     A
GRA                SX6    X1                                                     GRA      3714     A
GRA                LX6    IH.IHP                                                 GRA      3715     A
GRA                LX1    -IH.CAP                                                GRA      3716     A
GRA                LX2    -IH.CAP                                                GRA      3717     A
GRA                SX1    X1                                                     GRA      3718     A
GRA                SX2    X2                                                     GRA      3719     A
GRA                IX5    X1-X2                                                  GRA      3720     A
GRA                MX0    -IH.CAL                                                GRA      3721     A
GRA                BX4    -X0*X5                                                 GRA      3722     A
GRA                LX4    IH.CAP                                                 GRA      3723     A
GRA                MX0    -CC.H2L                                                GRA      3724     A
GRA                BX6    X4+X6       KEY = IHW(0,0, CA1-CA2 , IH1 )             GRA      3725     A
GRA                LX0    CC.H2P                                                 GRA      3726     A
GRA                SB2    7           REGN = 7                                   GRA      3727     A
GRA                SA5    RAT+7       RI = RAT(REGN)                             GRA      3728     A
GRA                                                                              GRA      3729     A
GRA      *         SEARCH B-REGISTERS FOR PREVIOUS OCCURANCE OF DIFF CANIDATE    GRA      3730     A
GRA                                                                              GRA      3731     A
GRA       SDC1     SA2    B6+X5                                                  GRA      3732     A
GRA                SA4    A2+B1                                                  GRA      3733     A
GRA                LX2    59-CA.ACP                                              GRA      3734     A
GRA                BX3    X4-X6                                                  GRA      3735     A
GRA                PL     X2,SDC2     IF ^AC[RCT(RI)]                            GRA      3736     A
GRA                NZ     X3,SDC2     IF RCT(RI+1) " KEY                         GRA      3737     A
GRA                SA4    A4+1                                                   GRA      3738     A
GRA                BX3    X7-X4                                                  GRA      3739     A
GRA                BX2    -X0*X3                                                 GRA      3740     A
GRA                ZR     X2,SDC      IF H2[RCT(RI+2)] = KEY2                    GRA      3741     A
GRA       SDC2     SB2    B2-B1       REGN = REGN - 1                            GRA      3742     A
GRA                SA5    A5-B1       RI = RAT(REGN)                             GRA      3743     A
GRA                NZ     B2,SDC1     IF REGN " 0                                GRA      3744     A
GRA                                                                              GRA      3745     A
GRA                EQ     SDC                                                    GRA      3746     A
GRA       AFA      TITLE  AFA - ASSIGN FULL ADDRESS TO A B-REG                   GRA      3747     A
GRA      **        AFA - ASSIGN FULL ADDRESS TO A B-REG                          GRA      3748     A
GRA      *                                                                       GRA      3749     A
GRA      *         ENTRY  (A0) = REGNO OF REGISTER WITH *RF*                     GRA      3750     A
GRA      *                (X5) = [BI] , B-CANIDATE TABLE WORD OF ENTRY           GRA      3751     A
GRA      *                (B6) = [O.RCT]                                         GRA      3752     A
GRA      *                                                                       GRA      3753     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA      *         EXIT    (A6) = CCW , ADDRESS OF CC WORD                    GRA      3754     A
GRA      *                 RAT, RCT UPDATED                                   GRA      3755     A
GRA                                                                           GRA      3756     A
GRA       AFA      ROUTINE                                                    GRA      3757     A
GRA                SA4     RAT+A0                                             GRA      3758     A
GRA                SA1     B6+X5                                              GRA      3759     A
GRA                SX3     X5          J = CO[BI]                             GRA      3760     A
GRA                LX3     RA.R3P                                             GRA      3761     A
GRA                BX6     X3+X4       R3[RAT(REGNO)] = J  */ SET TO APPEND IH,CA  GRA  3762     A
GRA                SA6     A4                                                 GRA      3763     A
GRA                MX0     1                                                  GRA      3764     A
GRA                SA2     A1+2        CCW = RCT(J+2)                         GRA      3765     A
GRA                LX0     1+CA.RAP                                           GRA      3766     A
GRA                BX7     X0+X1       RA[RCT(J)] = 1  */ INDICATE MODIFICATION  GRA  3767     A
GRA                SA7     A1                                                 GRA      3768     A
GRA                SX3     A0+2S6                                             GRA      3769     A
GRA                LX3     CC.REG1P    REG1[CCW] = REGNO                      GRA      3770     A
GRA                BX6     X2+X3       MIT[CCW] = 2  */ SLD, SST              GRA      3771     A
GRA                SA6     A2                                                 GRA      3772     A
GRA                EQ      AFA                                                GRA      3773     A
GRA       ECC      TITLE   ECC - ENTER CONSTANT CANIDATE                      GRA      3774     A
GRA      **        ECC - ENTER CONSTANT CANIDATE                              GRA      3775     A
GRA      *                                                                    GRA      3776     A
GRA      *         ENTRY   (X3) = CON                                         GRA      3777     A
GRA      *                 (X2) = CAW OF REF THAT IT IS PART OF               GRA      3778     A
GRA      *                 (B6) = [O.RCT]                                     GRA      3779     A
GRA                                                                           GRA      3780     A
GRA      *         BUMP OCCURANCES OF CANIDATE IN TABLE                       GRA      3781     A
GRA                                                                           GRA      3782     A
GRA       ECC0     SA2     A4-B1       CAW = RCT(I)                           GRA      3783     A
GRA                IX6     X2+X0       NOCC[CAW] = NOCC[CAW] + N              GRA      3784     A
GRA                SA6     A2                                                 GRA      3785     A
GRA                                                                           GRA      3786     A
GRA       ECC      ROUTINE                                                    GRA      3787     A
GRA                SA1     CL                                                 GRA      3788     A
GRA                MX0     -CA.NOCCL                                          GRA      3789     A
GRA                LX3     IH.CAP                                             GRA      3790     A
GRA                LX0     CA.NOCCP                                           GRA      3791     A
GRA                SB2     X1+1        I = CL                                 GRA      3792     A
GRA                BX0     -X0*X2      N = NOCC[CAW]                          GRA      3793     A
GRA                PL      X3,ECC1     KEY = IHW(0,0,ABS(CON),0)              GRA      3794     A
GRA                BX3     -X3                                                GRA      3795     A
GRA                                                                           GRA      3796     A
GRA       ECC1     SA4     B6+B2       C2W = RCT(I+1)                         GRA      3797     A
GRA                BX6     X3-X4                                              GRA      3798     A
GRA                ZR      X6,ECC0     IF [C2W] = KEY                         GRA      3799     A
GRA                SA1     A4+B1       CCW = RCT(I+2)                         GRA      3800     A
GRA                LX1     -CC.INCP                                           GRA      3801     A
GRA                SB2     X1+1        I = INC[CCW]                           GRA      3802     A
GRA                NE      B2,B1,ECC1  IF I " 0                               GRA      3803     A
GRA                                                                           GRA      3804     A
GRA      *         CON NOT ON CHAIN, ADD TO THE END OF RCT AND END OF CHAIN.  GRA      3805     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                                                                                GRA        3806     A
GRA                SA2    L.RCT       OL = L.RCT                                  GRA        3807     A
GRA                SX7    X2+3        L.RCT = L.RCT + 3                           GRA        3808     A
GRA                BX6    X2+X1       INC[CCW] = OL                               GRA        3809     A
GRA                SA7    A2                                                      GRA        3810     A
GRA                LX6    CC.INCP                                                 GRA        3811     A
GRA                SA6    A1                                                      GRA        3812     A
GRA                SX4    B1                                                      GRA        3813     A
GRA                LX4    CA.CONP                                                 GRA        3814     A
GRA                BX6    X0+X4       RCT(OL) = CAW(CON,0,N,0)                    GRA        3815     A
GRA                SA6    B6+X2                                                   GRA        3816     A
GRA                BX7    X3                                                      GRA        3817     A
GRA                SA7    A6+B1       RCT(OL+1) = KEY                             GRA        3818     A
GRA                MX6    0                                                       GRA        3819     A
GRA                SA6    A7+B1       RCT(OL+2) = 0                               GRA        3820     A
GRA                EQ     ECC                                                     GRA        3821     A
GRA       GRA      TTL    GRA - GLOBAL REGISTER ASSIGNMENT / ENTRY CODE           GRA        3822     A
GRA       SUP      TITLE  SUP - SETUP PRELOADS OF LOCKED REGISTERS                GRA        3823     A
GRA      **        TYPEI - DEFINE MACRO TO FORM A TYPE I INSTRUCTION              GRA        3824     A
GRA      *                J,K OPERANDS MUST BE IN X-REGISTERS.                   GRA        3825     A
GRA      *                R2 WORD IS ASSUMED TO BE IN X7 .                       GRA        3826     A
GRA                                                                               GRA        3827     A
GRA       TYPEI    MACRO  OPC,I,J,K                                               GRA        3828     A
GRA                R=     X6,I                                                    GRA        3829     A
GRA                IFC    NE,/J//,2                                               GRA        3830     A
GRA                L_J    R1.RJP                                                  GRA        3831     A
GRA                BX6    X6+J                                                    GRA        3832     A
GRA                IFC    NE,/K//,2                                               GRA        3833     A
GRA                L_K    R1.RKP                                                  GRA        3834     A
GRA                BX6    X6+K                                                    GRA        3835     A
GRA                IF     REG,OPC,2                                               GRA        3836     A
GRA                R=     B2,OPC                                                  GRA        3837     A
GRA                ELSE   1                                                       GRA        3838     A
GRA                R=     B2,OC.OPC                                               GRA        3839     A
GRA                RJ     SRI                                                     GRA        3840          I
          -CCGNC
GRA                RJ     =XSRI#                                                  CCGNC       657     A
GRA                ENDM                                                            GRA        3841     A
GRA       ORS      SPACE  2,14        ORS                                         GRA        3842     A
GRA      **        ORS - OUTPUT *RS*                                              GRA        3843     A
GRA      *                                                                         GRA        3844     A
GRA      *         ENTRY  (B5) = I , REGNO OF REGISTER                            GRA        3845     A
GRA                                                                               GRA        3846     A
GRA       ORS0     SB2    OC.RS                                                   GRA        3847     A
GRA                RJ     SRI         SRI(  RS  MI-4,FL+I )                       GRA        3848     A
GRA                                                                               GRA        3849     A
GRA       ORS      ROUTINE                                                        GRA        3850     A
GRA                SA2    RAT+B5      RW = RAT(I)                                 GRA        3851     A
GRA                SX4    B5+SO.LOCK                                              GRA        3852     A
GRA                SX6    B6-4                                                    GRA        3853     A
GRA                LX4    R1.SOP                                                  GRA        3854     A
GRA                BX6    X4+X6                                                   GRA        3855     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                MX4    1                                                    GRA        3856     A
GRA                BX7    X4*X2                                                GRA        3857     A
GRA                LX7    1+R1.SOP+SO.INVP                                     GRA        3858     A
GRA                BX6    X7+X6        INV[R1W] = INV[RAT(I)]                  GRA        3859     A
GRA                MX3    -2*RA.R1L                                            GRA        3860     A
GRA                SX7    B0           R2W = 0                                 GRA        3861     A
GRA                LX3    RA.R2P                                               GRA        3862     A
GRA                BX4    -X3*X2                                               GRA        3863     A
GRA                NZ     X4,ORS0      IF ( R2[RW] ! R3[RW] ) " 0              GRA        3864     A
GRA                                                                            GRA        3865     A
GRA                SA2    B4+X2        CAW = RCT(R1[RW])                       GRA        3866     A
GRA                SA3    A2+B1        C2W = RCT(R1[RW]+1)                     GRA        3867     A
GRA                SB2    X2                                                   GRA        3868     A
GRA                ZR     B2,ORS0      IF UDI[CAW] = 0  */ CON OR COMPILER TEMP GRA        3869     A
GRA                LX2    59-CA.ACP                                            GRA        3870     A
GRA                MI     X2,ORS0      IF AC[CAW]    */ PREFETCH               GRA        3871     A
GRA                MX4    -IH.CAIHL                                            GRA        3872     A
GRA                BX7    -X4*X3       R2W = IHW(0,0,CAIH[C2W])                GRA        3873     A
GRA                EQ     ORS0                                                 GRA        3874     A
GRA                                                                            GRA        3875     A
GRA        SUPA    BFMW   CA,(AC,CON,PL)                                       GRA        3876     A
GRA        SUP     SPACE  3,24         SUP                                    GRA        3877     A
GRA       **       SUP - SETUP PRELOADS OF LOCKED REGISTERS IN THE HOLDING BLOCK GRA        3878     A
GRA       *                                                                    GRA        3879     A
GRA       *        GENERAL REGISTER SETUP FOR *SUP* AND ITS SUBROUTINES        GRA        3880     A
GRA       *        (A0) = [O.SYM]-1 , (B4) = [O.RCT] , (B6,B7) = MI, MB        GRA        3881     A
GRA       *        AND X0 IS PRESERVED BY ALL THE SUBROUTINES                  GRA        3882     A
GRA                                                                            GRA        3883     A
GRA        SUP     ROUTINE                                                     GRA        3884     A
GRA                SX6    0            BSI = 0                                 GRA        3885     A
GRA                SA6    BSI                                                  GRA        3886     A
GRA                CALL   RTB          GET THE HOLDING BLOCK                   GRA        3887          I
          -CCGNC
GRA                CALL   RTB#         GET THE HOLDING BLOCK                   CCGNC       658     A
GRA                                                                            GRA        3888     A
GRA       *        INSERT THE *UXR* AND *MRA* FIELDS IN THE *R2* OF THE *BOS*  GRA        3889     A
GRA                                                                            GRA        3890     A
GRA                SA3    XRF                                                  GRA        3891     A
GRA                SA2    MRU                                                  GRA        3892     A
GRA                SA1    O.SEQ                                                GRA        3893     A
GRA                SA5    MRA                                                  GRA        3894     A
GRA                MX4    1                                                    GRA        3895     A
GRA                SA1    X1+B1        R2 = [O.SEQ+1]                          GRA        3896     A
GRA                BX6    X4*X3                                                GRA        3897     A
GRA                LX6    1+R2.UXRP    UXR[R2] = XRF & 1S59                    GRA        3898     A
GRA                BX2    -X2*X5                                               GRA        3899     A
GRA                LX2    R2.MRAP      MRA[R2] = ^MRU & MRA  */ MACHINE REGS AVAIL GRA        3900     A
GRA                BX7    X6+X2                                                GRA        3901     A
GRA                BX6    X7+X1                                                GRA        3902     A
GRA                SA6    A1                                                   GRA        3903     A
GRA                                                                            GRA        3904     A
GRA                ALLOC  MOD,16*17    ALLOC( MOD , 4*4*(7+7+3) )              GRA        3905     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA          SA3    O.SYM                                                GRA     3906    A
GRA          SA4    O.RCT                                                GRA     3907    A
GRA          SB7    X2          MB = [O.MOD]                             GRA     3908    A
GRA          SA0    X3-1                                                 GRA     3909        I
        -CCG1
GRA          SA0    X3+1                                                 CCG1     115    A
GRA          SB4    X4          (B4) = [O.RCT]                           GRA     3910    A
GRA          SA5    TRD                                                  GRA     3911    A
GRA          SB6    4           MI = 4        */ STORE INDEX             GRA     3912    A
GRA          SX6    X5-4                                                 GRA     3913    A
GRA          NZ     X6,SUP4     IF TRD " 4    */ NOT F(I), F(N) IN REGS  GRA     3914    A
GRA                                                                      GRA     3915    A
GRA     *    SETUP CODE TO INITIALIZE THE LOOP UPPER LIMIT REGISTER      GRA     3916    A
GRA                                                                      GRA     3917    A
GRA          SA1    LUL+1                                                GRA     3918    A
GRA          SA2    O.IIT                                                GRA     3919    A
GRA          SA5    RAT+X1      RW = RAT(LUL(2))                         GRA     3920    A
GRA          MX6    60                                                   GRA     3921    A
GRA          SA6    A5          RAT(LUL(2)) = -0                         GRA     3922    A
GRA          LX5    -RA.R2P     PFI = R2[RW]                             GRA     3923    A
GRA          SB2    X2                                                   GRA     3924    A
GRA          SA3    B2+X5       IAW = IIT(PFI)                           GRA     3925    A
GRA          UX6    B3,X3                                                GRA     3926    A
GRA          SX7    B3                                                   GRA     3927    A
GRA          IX0    X5+X7       PFI = PFI + NWD[IAW]                     GRA     3928    A
GRA          SA4    B2+X0       IAW = IIT(PFI)                           GRA     3929    A
GRA          SA5    A4+B1       IBW = IAW(PFI+1)                         GRA     3930    A
GRA          UX6    B3,X4                                                GRA     3931    A
GRA          SX1    B3-2        N = NWD[IAW] - 2  */ WORDS IN *IP* FORMULA GRA   3932    A
GRA          BX7    X1                                                   GRA     3933    A
GRA          SA7    N                                                    GRA     3934    A
GRA          ALLOC  MOD,X1      ALLOC( MOD , N)                          GRA     3935    A
GRA          SA3    O.SYM                                                GRA     3936    A
GRA          SA4    O.RCT                                                GRA     3937    A
GRA          SB7    X2          MB = [O.MOD]                             GRA     3938    A
GRA          SA0    X3-1                                                 GRA     3939        I
        -CCG1
GRA          SA0    X3+1                                                 CCG1     116    A
GRA          SB4    X4                                                   GRA     3940    A
GRA          SB6    4           MI = 4                                   GRA     3941    A
GRA          SA2    O.IIT                                                GRA     3942    A
GRA          SB2    X0+2        PFI = PFI + 2                            GRA     3943    A
GRA          SB5    B7                                                   GRA     3944    A
GRA          MOVE   X1,X2+B2,B7+B6    MOVE( N , O.IIT+PFI , MB+MI )      GRA     3945    A
GRA          SB7    B5                                                   GRA     3946    A
GRA          LX5    -IB.ILDP                                             GRA     3947    A
GRA          SB6    X5          MI = ILD[IBW]                            GRA     3948    A
GRA          LX0    RA.R2P                                               GRA     3949    A
GRA          SX5    X0          J = R1[RW]                               GRA     3950    A
GRA          RJ     LIV         INSERT LOAD OF *UL* IN FORMULA           GRA     3951    A
GRA          SA1    N                                                    GRA     3952    A
GRA          SA2    LCV+1                                                GRA     3953    A

        0         1         2         3         4         5         6         7         8
        12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SB6    X1+4        MI = N + 4                                    GRA    3954    A
GRA                SA3    RAT+X2                                                    GRA    3955    A
GRA                LX3    -RA.R3P                                                   GRA    3956    A
GRA                SX5    X3          J = R3[RAT(LUL(2))]                           GRA    3957    A
GRA                NZ     X5,SUP1     IF J " 0      */ ADDRESS APPENDED             GRA    3958    A
GRA                                                                                 GRA    3959    A
GRA                TYPEI  SA,B6,X1    SRI(  SA  MI,MI-4 )                           GRA    3960    A
GRA                EQ     SUP2                                                      GRA    3961    A
GRA                                                                                 GRA    3962    A
GRA        SUP1    SX4    X3+B1                                                     GRA    3963    A
GRA                SB3    X1                                                        GRA    3964    A
GRA                RJ     SIV         SIV( J , RF = N )                             GRA    3965    A
GRA                                                                                 GRA    3966    A
GRA        SUP2    SA1    LUL+1                                                     GRA    3967    A
GRA                SB5    X1+                                                       GRA    3968    A
GRA                RJ     ORS         ORS(  RS  MI-4,INVF )                         GRA    3969    A
GRA        SUP     SPACE  1,10                                                      GRA    3970    A
GRA       *        SCAN *RAT* AND SETUP INITIALIZATION CODE FOR THE B-REGISTERS     GRA    3971    A
GRA                                                                                 GRA    3972    A
GRA        SUP4    SB5    7           I = 7         */ REGNO                        GRA    3973    A
GRA                                                                                 GRA    3974    A
GRA        SUP5    SA1    RAT+B5                                                    GRA    3975    A
GRA                ZR     X1,SUP10    IF RAT(I) = 0 */ REGISTER NOT ASSIGNED        GRA    3976    A
GRA                                                                                 GRA    3977    A
GRA       *        EVALUATE THE INITIALIZATION FORMULA                              GRA    3978    A
GRA                                                                                 GRA    3979    A
GRA                BX0    X1          RW = RAT(I)                                   GRA    3980    A
GRA                SX5    X1          J = R1[RW]                                    GRA    3981    A
GRA                SA2    SUPA                                                      GRA    3982    A
GRA                MX6    -2*RA.R1L                                                 GRA    3983    A
GRA                SA3    B4+X1       CAW = RCT(J)                                  GRA    3984    A
GRA                LX0    -RA.R2P                                                   GRA    3985    A
GRA                BX7    -X6*X0                                                    GRA    3986    A
GRA                NZ     X7,SUP5A    IF (R2,R3)[RW] " 0                            GRA    3987    A
GRA                BX4    X2*X3                                                     GRA    3988    A
GRA                ZR     X4,SUP10    IF ^(CON,AC,PL)[CAW]  */ PRELOAD NOT NEC      GRA    3989    A
GRA                                                                                 GRA    3990    A
GRA        SUP5A   RJ     LIV         LIV(J)                                        GRA    3991    A
GRA                SX6    B6-4                                                      GRA    3992    A
GRA                SA6    RR          RR = MI-4                                     GRA    3993    A
GRA                SX5    X0          J = R2[RW]                                    GRA    3994    A
GRA                ZR     X5,SUP6     IF J = 0                                      GRA    3995    A
GRA                RJ     LIV         LIV( J )                                      GRA    3996    A
GRA                SA1    RR          ORN = RR                                      GRA    3997    A
GRA                SX2    B6-4                                                      GRA    3998    A
GRA                SX6    B6          RR = MI       */ OF *IS*                      GRA    3999    A
GRA                SA6    A1                                                        GRA    4000    A
GRA                TYPEI  IS,X6,X1,X2 SRI(  IS  MI,MI-4,ORN )                       GRA    4001    A
GRA                                                                                 GRA    4002    A
GRA        SUP6    LX0    RA.R2P-RA.R3P                                             GRA    4003    A
GRA                SX5    X0          J = R3[RW]                                    GRA    4004    A
GRA                ZR     X5,SUP7     IF J = 0                                      GRA    4005    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SA1    RR                                                GRA        4006    A
GRA                SX4    X0+B1                                             GRA        4007    A
GRA                SB3    X1                                                GRA        4008    A
GRA                RJ     SIV          SIV( J , RF = RR )                   GRA        4009    A
GRA                EQ     SUP9                                              GRA        4010    A
GRA                                                                         GRA        4011    A
GRA       *        TRANSFER THE RESULT TO A B-REGISTER , IF NECESSARY       GRA        4012    A
GRA                                                                         GRA        4013    A
GRA        SUP7    SX1    B6-4                                              GRA        4014    A
GRA                SA5    B7+X1        R1 = MB + MI-4                       GRA        4015    A
GRA                UX6    B2,X5                                             GRA        4016    A
GRA                SA4    A5+2         DI = R1 + 2                          GRA        4017    A
GRA                LX4    59-D.LDP                                          GRA        4018    A
GRA                MI     X4,SUP8      IF LD[DI]                            GRA        4019    A
GRA                SX6    B2-OC.IS                                          GRA        4020    A
GRA                NZ     X6,SUP9      IF OC " OC.IS                        GRA        4021    A
GRA                                                                         GRA        4022    A
GRA        SUP8    MX7    0                                                 GRA        4023    A
GRA                TYPEI  SA,B6,X1     SRI(  SA  MI,MI-4 )                  GRA        4024    A
GRA                                                                         GRA        4025    A
GRA        SUP9    RJ     ORS          ORS(  RS  MI-4,FL+I,INV )            GRA        4026    A
GRA                                                                         GRA        4027    A
GRA        SUP10   SB5    B5-B1        I = I - 1                            GRA        4028    A
GRA                NZ     B5,SUP5      IF I " 0                             GRA        4029    A
GRA        SUP     SPACE  1,14                                              GRA        4030    A
GRA       *        NOW SETUP PRELOADS FOR THE X-REGISTERS                   GRA        4031    A
GRA                                                                         GRA        4032    A
GRA                MX0    5                                                 GRA        4033    A
GRA                SB5    20B          I = 20B                              GRA        4034    A
GRA                LX0    -1           LDREG = 37BS54  */ LD-REG FLAG       GRA        4035    A
GRA                                                                         GRA        4036    A
GRA        SUP12   SA5    RAT+B5       J = RAT(I)    */ *R1* ONLY           GRA        4037    A
GRA                ZR     X5,SUP14     IF RAT(I) = 0                        GRA        4038    A
GRA                                                                         GRA        4039    A
GRA                SA1    B4+X5        CAW = RCT(J)                         GRA        4040    A
GRA                SA2    SUPA                                              GRA        4041    A
GRA                BX7    X2*X1                                             GRA        4042    A
GRA                ZR     X7,SUP14     IF ^(CON,PL,AC)[CAW]  */ PRELD NOT NECESS  GRA  4043    A
GRA                                                                         GRA        4044    A
GRA                RJ     LIV          LIV( J )                             GRA        4045    A
GRA                MI     X0,SUP13     IF LDREG < 0  */ RS IS TO A *LD* REG GRA        4046    A
GRA                SX1    B6-4                                              GRA        4047    A
GRA                TYPEI  XMT,B6,X1          SRI(  XMT  MI,MI-4 )           GRA        4048    A
GRA                                                                         GRA        4049    A
GRA        SUP13   RJ     ORS          ORS(  RS  MI-4,FL+I,INV )            GRA        4050    A
GRA                                                                         GRA        4051    A
GRA        SUP14   SB5    B5+B1        I = I + 1                            GRA        4052    A
GRA                LX0    1            LDREG = SHIFT(LDREG,1)               GRA        4053    A
GRA                SX6    B5-30B                                            GRA        4054    A
GRA                MI     X6,SUP12     IF I @ 27B                           GRA        4055    A
GRA        SUP     SPACE  2,14                                              GRA        4056    A
GRA       *        SETUP *MOD* CONTROL WORD AND MERGE THIS WITH *HB*        GRA        4057    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                                                                                GRA        4058     A
GRA                SA1    L.SEQ                                                   GRA        4059     A
GRA                SX0    X1-8        II = L.SEQ-8                                GRA        4060     A
GRA                SX6    B6                                                      GRA        4061     A
GRA                LX0    ML.IIP                                                  GRA        4062     A
GRA                SA6    L.MOD       L.MOD = MI                                  GRA        4063     A
GRA                SX2    4                                                       GRA        4064     A
GRA                IX6    X6-X2                                                   GRA        4065     A
GRA                LX6    ML.NIP-2    NI = (MI-4) / 4                             GRA        4066     A
GRA                LX2    ML.MTIP                                                 GRA        4067     A
GRA                BX3    X0+X2                                                   GRA        4068     A
GRA                IX1    X3+X6                                                   GRA        4069     A
GRA                SA5    O.SEQ       HBA = O.SEQ                                 GRA        4070     A
GRA                ZR     X6,SUP15    IF NI = 0     */ NO MODS                    GRA        4071     A
GRA                                                                               GRA        4072     A
GRA                ADDWRD MLT,X1      ADDWRD( MLT , MCW(0,II,NI,4) )              GRA        4073     A
GRA                MX6    1                                                       GRA        4074     A
GRA                CALL   MPB         MERGE MODS WITH *HB*, SQUEEZE AND REWRITE   GRA        4075          I
          -CCGNC
GRA                CALL   MPB#        MERGE MODS WITH *HB*, SQUEEZE AND REWRITE   CCGNC       659     A
GRA                                                                               GRA        4076     A
GRA                SA5    O.TXT       HBA = O.TXT                                 GRA        4077     A
GRA                                                                               FCC0448     108          I
          -FCC0675
GRA       *        SCAN *BST*, UPDATE *LBN* TO ACCOUNT FOR INNER LOOPS.            FCC0448     109          I
          -FCC0675
GRA                                                                               FCC0448     110          I
          -FCC0675
GRA        SUP15   SA4    O.BST                                                   FCC0448     111          I
          -FCC0675
GRA                SA2    O.BIT                                                   FCC0448     112          I
          -FCC0675
GRA                MX0    -BI.LCBL                                                FCC0448     113          I
          -FCC0675
GRA                SA1    LBN                                                     FCC0448     114          I
          -FCC0675
GRA                LX0    BI.LCBP                                                 FCC0448     115          I
          -FCC0675
GRA                SB3    X2                       (B3) = O.BIT                   FCC0448     116          I
          -FCC0675
GRA                LX1    BI.LCBP                                                 FCC0448     117          I
          -FCC0675
GRA                SA3    X4+2        BAW = BST(2)                                FCC0448     118          I
          -FCC0675
GRA                                                                               FCC0448     119          I
          -FCC0675
GRA        SUP15A  LX3    -BA.BIP     BN = BI[BAW]                                FCC0448     120          I
          -FCC0675
GRA                SA2    B3+X3       BIW = BIT(BN)                               FCC0448     121          I
          -FCC0675
GRA                LX2    59-BI.RBP                                               FCC0448     122          I
          -FCC0675

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                 MI     X2,SUP15B   IF RB[BIW]    */ ^ A *HB*                          FCC0448      123          I
          -FCC0675
GRA                 LX2    1+BI.RBP-BI.PBNP  J = PBN[BIW]  */ BN OF HEADER                FCC0448      124          I
          -FCC0675
GRA                 SA2    B3+X2       BIW = BIT(J)                                        FCC0448      125          I
          -FCC0675
GRA                 BX6    -X0*X2                                                          FCC0448      126          I
          -FCC0675
GRA                 IX7    X1-X6                                                           FCC0448      127          I
          -FCC0675
GRA                 PL     X7,SUP15B   LBN = MAX( LBN , LCB[BIW] )                         FCC0448      128          I
          -FCC0675
GRA                 BX1    X6                                                              FCC0448      129          I
          -FCC0675
GRA                                                                                        FCC0448      130          I
          -FCC0675
GRA        SUP15B   SA3    A3+2        BAW = BAW + 2                                       FCC0448      131          I
          -FCC0675
GRA                 NZ     X3,SUP15A   IF [BAW] " 0  */ NOT END OF *BST*                   FCC0448      132          I
          -FCC0675
GRA                                                                                        FCC0448      133          I
          -FCC0675
GRA                 SA3    X4+2                                                            FCC0448      134          I
          -FCC0675
GRA                 LX3    -BA.BIP     BN = BI[BST(2)]                                     FCC0448      135          I
          -FCC0675
GRA                 SA2    B3+X3                                                           FCC0448      136          I
          -FCC0675
GRA                 BX6    X0*X2                                                           FCC0448      137          I
          -FCC0675
GRA                 BX7    X1+X6       LCB[BIT(BN)] = LBN  */ SET *LBN* OF HEADER          FCC0448      138          I
          -FCC0675
GRA                 SA7    A2                                                              FCC0448      139          I
          -FCC0675
GRA                                                                                        GRA         4078          I
          -FCC0675
GRA       *         SCAN HOLDING BLOCK FOR *TST*"S AND UPDATE THEIR *LU* FIELD              GRA         4079          I
          -FCC0675
GRA       *         FOR LATER USE IN *POST* DURING THE TEMPORARY PACKING.                   GRA         4080          I
          -FCC0675
GRA                                                                                        GRA         4081          I
          -FCC0675
GRA        SUP15    SA2    O.TET                                                           GRA         4082          I
          -FCC0448  -FCC0675
GRA                 SA1    LBN                                                             GRA         4083          I
          -FCC0448  -FCC0675
GRA                 MX0    -T.LUL                                                          GRA         4084          I
          -FCC0448  -FCC0675
GRA                 SA1    LBN                                                             GRA         4085          I
          -FCC0448  -FCC0675
GRA                 SA2    O.TET                                                           FCC0448      140          I
          -FCC0675

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                 LX1    T.LUP-BI.LCBP                                    FCC0448      141          I
          -FCC0675
GRA                 MX0    -T.LUL                                           FCC0448      142          I
          -FCC0675
GRA                 LX0    T.LUP                                            GRA         4086          I
          -FCC0675
GRA                 SB4    4                                                GRA         4087          I
          -FCC0675
GRA                 SA5    X5+B4       R1 = HBA + 4                         GRA         4088          I
          -FCC0675
GRA                 SB5    X2                                               GRA         4089          I
          -FCC0675
GRA                 SB3    OC.TST                                           GRA         4090          I
          -FCC0675
GRA                 SA2    ITL                                              GRA         4091          I
          -FCC0675
GRA                 LX1    T.LUP                                            GRA         4092          I
          -FCC0448  -FCC0675
GRA                 SB7    X2                        (B7) = ITL             GRA         4093          I
          -FCC0675
GRA                                                                         GRA         4094          I
          -FCC0675
GRA        SUP16    UX6    B2,X5       OC = OC[R1]                          GRA         4095          I
          -FCC0675
GRA                 SA5    A5+B4       R1 = R1 + 4                          GRA         4096          I
          -FCC0675
GRA                 ZR     B2,SUP      IF OC = OC.EOQ  */ END OF BLOCK      GRA         4097          I
          -FCC0675
GRA                 NE     B2,B3,SUP16 IF OC " OC.TST                       GRA         4098          I
          -FCC0675
GRA                                                                         GRA         4099          I
          -FCC0675
GRA                 SA4    A5-3        R2 = R1 - 3                          GRA         4100          I
          -FCC0675
GRA                 LX4    -IH.CAP     C = CA[R2]                           GRA         4101          I
          -FCC0675
GRA                 SA3    B5+X4                                            GRA         4102          I
          -FCC0675
GRA                 BX6    X1          L = LBN                              GRA         4103          I
          -FCC0675
GRA                 SB6    X4                                               GRA         4104          I
          -FCC0675
GRA                 GE     B6,B7,SUP17 IF C \ ITL    */ CREATED IN THIS REGION  GRA    4105          I
          -FCC0675
GRA                 BX2    -X0*X3                                           GRA         4106          I
          -FCC0675
GRA                 IX7    X6-X2                                            GRA         4107          I
          -FCC0675
GRA                 PL     X7,SUP17    L = MAX( LU[TET(C)] , LBN )          GRA         4108          I
          -FCC0675
GRA                 BX6    X2                                               GRA         4109          I
          -FCC0675

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                                                                                GRA        4110        I
          -FCC0675
GRA        SUP17    BX2    X0*X3                                                  GRA        4111        I
          -FCC0675
GRA                 IX6    X6+X2       LU[TET(C)] = L                             GRA        4112        I
          -FCC0675
GRA                 SA6    A3                                                     GRA        4113        I
          -FCC0675
GRA                 EQ     SUP16                                                  GRA        4114        I
          -FCC0675
GRA                                                                               FCC0675     179     A
GRA       *         SCAN TET AND CLEAR *HBN* FIELD OF IT."S CREATED IN THIS LOOP   FCC0675     180     A
GRA                                                                               FCC0675     181     A
GRA        SUP15    SA1    O.TET                                                  FCC0675     182     A
GRA                 SA2    ITL                                                    FCC0675     183     A
GRA                 SA3    L.TET                                                  FCC0675     184     A
GRA                 SB2    X2          I = ITL                                    FCC0675     185     A
GRA                 SB6    X3          L = L.TET                                  FCC0675     186     A
GRA                 MX0    -T.HBNL                                                FCC0675     187     A
GRA                 SB5    X1                      (B5) = TET                     FCC0675     188     A
GRA                 LX0    T.HBNP                                                 FCC0675     189     A
GRA                 SB7    X2                      (B7) = ITL                     FCC0675     190     A
GRA                 EQ     B2,B6,SUP17 IF ITL = L.TET  */ NO IT. S CREATED        FCC0675     191     A
GRA                                                                               FCC0675     192     A
GRA        SUP16    SA2    B5+B2                                                  FCC0675     193     A
GRA                 BX6    X0*X2       HBN[TET(I)] = 0                            FCC0675     194     A
GRA                 SB2    B2+B1       I = I + 1                                  FCC0675     195     A
GRA                 SA6    A2                                                     FCC0675     196     A
GRA                 LT     B2,B6,SUP16 IF I < L                                   FCC0675     197     A
GRA                                                                               FCC0675     198     A
GRA       *         SCAN *HB* AND SET *HBN* FIELD OF IT. S DEFINED IN THIS LOOP    FCC0675     199     A
GRA                                                                               FCC0675     200     A
GRA        SUP17    SA1    =XHBI                                                  FCC0675     201     A
GRA                 SB4    4                                                      FCC0675     202     A
GRA                 SA5    X5+B4       R1 = HBA + 4                               FCC0675     203     A
GRA                 SB3    OC.TST                                                 FCC0675     204     A
GRA                 LX1    T.HBNP                                                 FCC0675     205     A
GRA                                                                               FCC0675     206     A
GRA        SUP18    UX6    B2,X5       OC = OC[R1]                                FCC0675     207     A
GRA                 SA5    A5+B4       R1 = R1 + 4                                FCC0675     208     A
GRA                 ZR     B2,SUP19    IF OC = OC.EOQ  */ END OF BLOCK            FCC0675     209     A
GRA                 NE     B2,B3,SUP18 IF OC " OC.TST                             FCC0675     210     A
GRA                                                                               FCC0675     211     A
GRA                 SA4    A5-3        R2 = R1 - 3                                FCC0675     212     A
GRA                 LX4    -IH.CAP     C = CA[R2]                                 FCC0675     213     A
GRA                 SA3    B5+X4                                                  FCC0675     214     A
GRA                 BX2    X0*X3                                                  FCC0675     215     A
GRA                 IX6    X1+X2       HBN[TET(C)] = HBI                          FCC0675     216     A
GRA                 SA6    A3                                                     FCC0675     217     A
GRA                 EQ     SUP18                                                  FCC0675     218     A
GRA                                                                               FCC0675     219     A
GRA       *         SCAN *TET* BACKWARDS AND REMOVE TRAILING ENTRIES THAT WERE NOT FCC0675     220     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA      *         MATERIALIZED ( ASSIGNED TO A REGISTER, KD, ETC ).                FCC0675     221    A
GRA                                                                                FCC0675     222    A
GRA       SUP19    SB6    B6-B1       L = L - 1                                    FCC0675     223    A
GRA                SA4    B5+B6                                                    FCC0675     224    A
GRA                LT     B6,B7,SUP20 IF L < ITL                                   FCC0675     225    A
GRA                BX6    -X0*X4                                                   FCC0675     226    A
GRA                ZR     X6,SUP19    IF HBN[TET(L)] = 0  */ NOT MATERIALIZED      FCC0675     227    A
GRA                                                                                FCC0675     228    A
GRA       SUP20    SA1    =XN.GT                                                   FCC0675     229    A
GRA                SX6    B6+B1       L.TET = L + 1                                FCC0675     230    A
GRA                IX2    X1-X6                                                    FCC0675     231    A
GRA                SA6    L.TET                                                    FCC0675     232    A
GRA                MI     X2,SUP      N.GT = MIN( N.GT , L.TET )                   FCC0675     233    A
GRA                SA6    A1                                                       FCC0675     234    A
GRA                EQ     SUP                                                      FCC0675     235    A
GRA       LIV      SPACE  3,24        LIV                                          GRA        4115    A
GRA      **        LIV - LOAD INITIAL VALUE                                        GRA        4116    A
GRA      *                                                                         GRA        4117    A
GRA      *         ENTRY  (X5) = J , *RCT* ORDINAL OF CANIDATE                     GRA        4118    A
GRA                                                                                GRA        4119    A
GRA       LIV      ROUTINE                                                         GRA        4120    A
GRA                SA1    B4+X5       CAW = RCT(J)                                 GRA        4121    A
GRA                SA2    A1+B1       C2W = RCT(J+1)                               GRA        4122    A
GRA                MX7    -IH.CAL                                                  GRA        4123    A
GRA                LX1    59-CA.CONP                                               GRA        4124    A
GRA                PL     X1,LIV2     IF ^CON[CAW]                                 GRA        4125    A
GRA                                                                                GRA        4126    A
GRA                SB2    OC.S                                                     GRA        4127    A
GRA                LX7    IH.CAP                                                   GRA        4128    A
GRA                BX6    -X7*X2                                                   GRA        4129    A
GRA                MX7    0                                                        GRA        4130    A
GRA                LX6    R1.INP-IH.CAP                                            GRA        4131    A
GRA                LX2    59-IH.MSKP                                               GRA        4132    A
GRA                PL     X2,LIV1     IF ^MSK[C2W]                                 GRA        4133    A
GRA                SB2    OC.FMA                                                   GRA        4134    A
GRA                NZ     X6,LIV1     IF IN[R1] " 0                                GRA        4135    A
GRA                SB2    OC.CLR                                                   GRA        4136    A
GRA                                                                                GRA        4137    A
GRA       LIV1     SX4    B6                                                       GRA        4138    A
GRA                BX6    X4+X6                                                    GRA        4139    A
GRA                RJ     SRI         SRI(  S/FMA/CLR  CA[C2W],MI )                GRA        4140    A
GRA                EQ     LIV                                                      GRA        4141    A
GRA                                                                                GRA        4142    A
GRA       LIV2     LX1    CA.CONP-CA.VCP                                           GRA        4143    A
GRA                MI     X1,LIV3     IF VC[CAW]                                   GRA        4144    A
GRA                                                                                GRA        4145    A
GRA                SB3    B0          RF = 0                                       GRA        4146    A
GRA                SX4    X5+B1                                                    GRA        4147    A
GRA                RJ     SIV         SIV( J )                                     GRA        4148    A
GRA                EQ     LIV                                                      GRA        4149    A
GRA                                                                                GRA        4150    A
GRA      *         *VC* - OUPUT *LD* OF INITIAL VALUE                              GRA        4151    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                                                                    GRA        4152     A
GRA       LIV3     LX2     -IH.RFP                                     GRA        4153     A
GRA                SB3     X2          R = RF[C2W]                     GRA        4154     A
GRA                LE      B3,B1,LIV4  IF R @ 1                        GRA        4155     A
GRA                SX4     B3                                          GRA        4156     A
GRA                SB3     B0                                          GRA        4157     A
GRA                RJ      OIL         OIL( J = RF[C2W] , RF = 0 )     GRA        4158     A
GRA                SB3     B6-4        RF = MI - 4                     GRA        4159     A
GRA                                                                    GRA        4160     A
GRA       LIV4     SX4     X5                                          GRA        4161     A
GRA                RJ      OIL         OIL( J , R )                    GRA        4162     A
GRA                EQ      LIV                                         GRA        4163     A
GRA       OIL      SPACE   3,18        OIL                             GRA        4164     A
GRA      **        OIL - OUTPUT INITIAL *LD*                           GRA        4165     A
GRA      *                                                             GRA        4166     A
GRA      *         ENTRY   (X4) = J , *RCT* ORDINAL OF CANIDATE        GRA        4167     A
GRA      *                 (B3) = RF , OPERAND R-NUMBER                GRA        4168     A
GRA      *                                                             GRA        4169     A
GRA      *         PRESERVES X0, X5                                    GRA        4170     A
GRA                                                                    GRA        4171     A
GRA       OIL      ROUTINE                                             GRA        4172     A
GRA                SA1     B4+X4       CAW = RCT(J)                    GRA        4173     A
GRA                SA2     A1+B1       C2W = RCT(J+1)                  GRA        4174     A
GRA                SX7     B3                                          GRA        4175     A
GRA                MX3     -IH.CAIHL                                   GRA        4176     A
GRA                LX7     IH.RFP                                      GRA        4177     A
GRA                BX2     -X3*X2      CAIH = CAIH[[C2W]               GRA        4178          I
          -CCG16
GRA                IX7     X7+X2       R2W = IHW(0,RF,CAIH[C2W])       GRA        4179          I
          -CCG16
GRA                BX3     -X3*X2      CAIH = CAIH[C2W]                CCG16       219     A
GRA                IX7     X7+X3       R2W = IHW(0,RF,CAIH)            CCG16       220     A
GRA                SX6     X1          U = UDI[CAW]                    GRA        4180     A
GRA                SB2     OC.LD       OC = OC.LD                      GRA        4181          I
          -CCG16
GRA                AX2     IH.SLVP                                     CCG16       221     A
GRA                SA3     OILA+X2                                     CCG16       222     A
GRA                SB2     X3          OC = OILA(SLV[C2W])             CCG16       223     A
GRA                EQ      B3,B1,OIL5  IF RF = 1                       GRA        4182     A
GRA                LX1     59-CA.RAP                                   GRA        4183     A
GRA                PL      X1,OIL2     IF ^RA[CAW]   */ NOT ASSIGNED TO A REG  GRA   4184     A
GRA                NZ      X2,OIL2     IF LSV[C2W] " 0  */ LDV OR LDV  CCG16       224          I
          -CCG21
GRA                NZ      X2,OIL2     IF LSV[C2W] " 0  */ LDC OR LDV  CCG21        67     A
GRA                NZ      B3,OIL2     IF RF " 0                       GRA        4185     A
GRA                                                                    GRA        4186     A
GRA      *         CHECK VARIABLE AND SETUP AN *ILD* IF DEAD ON EXIT FROM LOOP  GRA  4187  A
GRA                                                                    GRA        4188     A
GRA                SB2     OC.ILD                                      GRA        4189     A
GRA                ZR      X6,OIL1     IF U = 0      */ COMPILER TEMPORARY  GRA   4190     A
GRA                BX3     X1                                          GRA        4191          I
          -FCC1646X

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SA3    O.UDT                                                    FCC1646X       1     A
GRA                SB3    X6                                                       FCC1646X       2     A
GRA                SA4    X3+B3       UI = UDT(U)                                  FCC1646X       3     A
GRA                LX4    59-UD.CMP                                                FCC1646X       4     A
GRA                PL     X4,OIL0     IF ^CM[UI]                                   FCC1646X       5     A
GRA                                                                                FCC1646X       6     A
GRA      *         *CR*, DO NOT OUTPUT A *ILD* IF THE BASE IS USED IN THE LOOP.    FCC1646X       7     A
GRA                                                                                FCC1646X       8     A
GRA                LX4    1+UD.CMP-UD.BMIP                                         FCC1646X       9     A
GRA                SB3    X3+B1                                                    FCC1646X      10     A
GRA                SA4    B3+X4       UB = UDT(BMI[UI]+1)                          FCC1646X      11     A
GRA                SA3    UVA                                                      FCC1646X      12     A
GRA                SB3    X4                                                       FCC1646X      13     A
GRA                SA3    X3+B3       UVW = UVA(WI[UB])                            FCC1646X      14     A
GRA                UX4    B3,X4                                                    FCC1646X      15     A
GRA                SB3    B3-59                                                    FCC1646X      16     A
GRA                LX3    -B3                                                      FCC1646X      17     A
GRA                MI     X3,OIL1A    IF SHIFT(UVW,59-BN) < 0  */ BASE USED        FCC1646X      18     A
GRA                                                                                FCC1646X      19     A
GRA       OIL0     BX3    X1                                                       FCC1646X      20     A
GRA                LX3    CA.RAP-CA.LXP                                            GRA         4192     A
GRA                PL     X3,OIL2     IF ^LX[CAW]   */ DEAD ON EXIT                GRA         4193     A
GRA                LX3    CA.LXP-CA.PSPP                                           GRA         4194     A
GRA                MI     X3,OIL2     IF PSP[CAW]   */ POST STORE POSSIBLE         GRA         4195     A
GRA                SB2    OC.LD       OC = OC.LD                                   GRA         4196          I
          -FCC1646X
GRA       OIL1A    SB2    OC.LD       OC = OC.LD                                   FCC1646X      21     A
GRA                EQ     OIL2                                                     GRA         4197     A
GRA                                                                                GRA         4198     A
GRA       OIL1     LX2    -IH.CAP                                                  GRA         4199     A
GRA                SA3    ITL                                                      GRA         4200     A
GRA                SX2    X2                                                       GRA         4201     A
GRA                IX4    X2-X3                                                    GRA         4202     A
GRA                PL     X4,OIL2     IF CA[C2W] \ ITL  */ CREATED IN THIS LOOP    GRA         4203     A
GRA                SB2    OC.TLD      OC = OC.TLD                                  GRA         4204     A
GRA                                                                                GRA         4205     A
GRA       OIL2     SX2    B6                                                       GRA         4206     A
GRA                LX6    R1.INP                                                   GRA         4207     A
GRA                BX6    X2+X6                                                    GRA         4208     A
GRA                LX1    CA.RAP-CA.ACP                                            GRA         4209     A
GRA                PL     X1,OIL3     IF ^AC[CAW]   */ NOT A PREFETCH              GRA         4210     A
GRA                                                                                GRA         4211     A
GRA                MX1    1                                                        GRA         4212     A
GRA                LX1    1+IH.LDP                                                 GRA         4213     A
GRA                BX7    X1+X7       LD[R2W] = 1   */ INHIBIT SQUEEZING           GRA         4214     A
GRA                                                                                GRA         4215     A
GRA       OIL3     SB3    X7                                                       GRA         4216          I
          -FCC0882
GRA       OIL3     SB3    X7-1S12                                                  FCC0882        7     A
GRA                SX1    0           DB = 0                                       FCC0882        8     A
GRA                PL     B3,OIL4     IF IH[R2W] > 1S12  */ LVL2 OR SUCH           FCC0882        9     A
GRA                SB3    X7                                                       FCC0882       10     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SB3    B3+B3                                                GRA        4217        I
          -CCG1
GRA                SA2    A0-B3       WORDB = SYM(IH[R2W])                     GRA        4218        I
          -CCG1
GRA                LX2    -P.FPB                                               GRA        4219        I
          -CCG1
GRA                SX2    B3+B3                                                CCG1        117    A
GRA                SB3    A0+B3                                                CCG1        118    A
GRA                SA2    B3+X2       WORDB = SYM(3*IH[R2W]+1)                 CCG1        119    A
GRA                LX2    -WB.FPP                                              CCG1        120    A
GRA                SX3    B1                                                   GRA        4220    A
GRA                BX1    X3*X2                                                GRA        4221    A
GRA                LX1    D.FPP       FP[DB] = FP[WORDB]                       GRA        4222    A
GRA                                                                            GRA        4223    A
GRA       OIL4     SA4    F.RDT+B2                                             GRA        4224    A
GRA                PX6    B2,X6                                                GRA        4225    A
GRA                SA6    B7+B6       MOD(MI) = TYII(OC,U,0,MI)                GRA        4226    A
GRA                SA7    A6+B1       MOD(MI+1) = R2W                          GRA        4227    A
GRA                BX6    X4-X1       MOD(MI+2) = XOR(RDT(OC),DB)              GRA        4228    A
GRA                SA6    A7+B1                                                GRA        4229    A
GRA                MX7    0                                                    GRA        4230    A
GRA                SA7    A6+B1       MOD(MI+3) = 0                            GRA        4231    A
GRA                SB6    B6+4        MI = MI + 4                              GRA        4232    A
GRA                EQ     OIL                                                  GRA        4233    A
GRA                                                                            GRA        4234    A
GRA       *        SPECIAL CASE, RF = 1 AND IH = 0                             GRA        4235    A
GRA                                                                            GRA        4236    A
GRA       OIL5     SX4    B6                                                   GRA        4237    A
GRA                LX6    R1.INP                                               GRA        4238    A
GRA                MX1    0           DB = 0                                   GRA        4239    A
GRA                BX6    X4+X6                                                GRA        4240    A
GRA                NZ     X2,OIL4     IF CAIH " 0   */ CA .NE. 0, SINCE IH = 0 GRA        4241    A
GRA                MX1    2                                                    GRA        4242    A
GRA                LX1    2+D.SZP     SZ[DB] = 3    */ SET TO TOGGLE *SZ* FIELD GRA        4243    A
GRA                EQ     OIL4                                                 GRA        4244    A
GRA                                                                            CCG16       225    A
GRA       OILA     CON    OC.LD,OC.LDC,OC.LDV                                  CCG16       226    A
GRA       SIV      TITLE  SIV - SET INITIAL VALUE                              GRA        4245    A
GRA      **        SIV - SET INITIAL VALUE, GENERATE A *STT* OF A VALUE        GRA        4246    A
GRA      *         NOTE THAT THE *RF* IN THE *RCT* ENTRY IS IGNORED.           GRA        4247    A
GRA      *                                                                     GRA        4248    A
GRA      *         ENTRY  (X4) = J+1, J = *RCT* ORDINAL OF ENTRY               GRA        4249    A
GRA      *                (B3) = RF , OPERAND R-NUMBER                         GRA        4250    A
GRA                                                                            GRA        4251    A
GRA      *         OUTPUT STT WITH AN *H2* FIELD                               GRA        4252    A
GRA                                                                            GRA        4253    A
GRA       SIV0     SX6    X2                                                   GRA        4254    A
GRA                SX7    B3                                                   GRA        4255    A
GRA                RJ     OSI         OSI(  STT  R,RF,CAIH,H2 )                GRA        4256    A
GRA                                                                            GRA        4257    A
GRA       SIV      ROUTINE                                                     GRA        4258    A
GRA                SA1    B4+X4       C2W = RCT(J+1)                           GRA        4259    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                 SA2    A1+B1       CCW = RCT(J+2)                                GRA          4260     A
GRA                 MX7    -IH.CAIHL                                                 GRA          4261     A
GRA                 LX2    -CC.H2P                                                   GRA          4262     A
GRA                 BX1    -X7*X1      CAIH = CAIH[C2W]                              GRA          4263     A
GRA                 SB2    X2          H2 = H2[CCW]                                  GRA          4264     A
GRA                 ZR     B2,SIV0     IF H2 = 0                                     GRA          4265     A
GRA                                                                                  GRA          4266     A
GRA       *         CHECK ADDRESS DIFFERENCE TO SEE IF SYMS ARE F.P. OR COMMON       GRA          4267     A
GRA                                                                                  GRA          4268     A
GRA                 SB2    B2+B2                                                     GRA          4269          I
          -CCG1
GRA                 LX6    B1,X1                                                     GRA          4270          I
          -CCG1
GRA                 SA3    A0-B2       WORDB2 = SYM(H2)                              GRA          4271          I
          -CCG1
GRA                 SB2    X6                                                        GRA          4272          I
          -CCG1
GRA                 ZR     B2,SIV1     IF IH[C2W] = 0  */ NEG RELOC NOT ALLOWED      FCC0645        21          I
          -CCG1
GRA                 SA4    A0-B2       WORDB = SYM(IH[C2W])                          GRA          4273          I
          -CCG1
GRA                 BX6    X3+X4                                                     GRA          4274          I
          -CCG1
GRA                 LX6    59-P.FPB                                                  GRA          4275          I
          -CCG1
GRA                 MI     X6,SIV1     IF FP[WORDB] ! FP[WORDB2]                     GRA          4276          I
          -CCG1
GRA                 LX6    1+P.FPB+58-P.RL                                           GRA          4277          I
          -CCG1
GRA                 BX7    X3-X4                                                     GRA          4278          I
          -CCG1
GRA                 PL     X6,SIV0     IF ^( COM[WORDB] ! COM[WORDB2] )              GRA          4279          I
          -CCG1
GRA                 SB2    B2+B2                                                     CCG1          121     A
GRA                 LX6    B1,X1                                                     CCG1          122     A
GRA                 SB2    X2+B2                                                     CCG1          123     A
GRA                 SA3    A0+B2       WORDB2 = SYM(3*H2+1)                          CCG1          124     A
GRA                 SB2    X6                                                        CCG1          125     A
GRA                 ZR     B2,SIV1     IF IH[C2W]  = 0  */ NEG RELOC NOT ALLOWED     CCG1          126     A
GRA                 SB2    B2+X1                                                     CCG1          127     A
GRA                 SA4    A0+B2       WORDB = SYM(3*IH[C2W]+1)                      CCG1          128     A
GRA                 BX6    X3+X4                                                     CCG1          129     A
GRA                 LX6    59-WB.FPP                                                 CCG1          130          I
          -CCG16
GRA                 LX6    59-WB.LABP                                                CCG16         227     A
GRA                 MI     X6,SIV0     IF LAB[WORDB] ! LAB[WORDB2]  */ LAB DIFF      CCG16         228     A
GRA                 LX6    WB.LABP-WB.FPP                                            CCG16         229     A
GRA                 MI     X6,SIV1     IF FP[WORDB] ! FP[WORDB2]                     CCG1          131     A
GRA                 LX6    1+WB.FPP-WB.CXP                                           CCG1          132     A
GRA                 MX7    -WB.CXL                                                   CCG1          133     A
GRA                 BX6    -X7*X6                                                    CCG1          134     A
GRA                 ZR     X6,SIV0     IF CX[WORDB] ! CX[WORDB2]                     CCG1          135          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

          -CCG16
GRA                ZR     X6,SIV0     IF ^( CX[WORDB] ! CX[WORDB2] )                  CCG16        230     A
GRA                SA3    A3+B1       WORDC2 = WORDB2 + 1                             CCG16        231     A
GRA                SA4    A4+B1       WORDC = WORDB + 1                               CCG16        232     A
GRA                BX7    X3-X4                                                       CCG1         136     A
GRA                SA4    SIVA                                                        GRA         4280     A
GRA                BX6    X4*X7                                                       GRA         4281     A
GRA                ZR     X6,SIV0     IF (RL,RB)[WORDB] = (RL,RB)[WORDB2]             GRA         4282          I
          -CCG16
GRA                ZR     X6,SIV0     IF (RL,RB)[WORDC] = (RL,RB)[WORDC2]             CCG16        233     A
GRA                                                                                   GRA         4283     A
GRA                MX3    0           C = 0                                           GRA         4284     A
GRA                EQ     SIV2                                                        GRA         4285     A
GRA                                                                                   GRA         4286     A
GRA       SIV1     BX3    X1          C = CA[C2W]                                     GRA         4287     A
GRA                SX1    X1          CAIH = IH[C2W]                                  GRA         4288     A
GRA                BX3    X3-X1                                                       GRA         4289     A
GRA                                                                                   GRA         4290     A
GRA       SIV2     MX7    0           F = 0                                           GRA         4291     A
GRA                SX6    B0          H2 = 0                                          GRA         4292     A
GRA                RJ     OSI         OSI(  STT  R,,CAIH )                            GRA         4293     A
GRA                SX1    X2          CAIH = H2[CCW]                                  GRA         4294     A
GRA                SX6    B0          H2 = 0                                          GRA         4295     A
GRA                SX7    0           F = 0                                           GRA         4296          I
          -FCC0912
GRA                MX7    0           F = 0                                           FCC0912        7     A
GRA                BX2    X5          FLB = FPL2    */ SAVE                           FCC0912        8     A
GRA                RJ     OSI         OSI(  STT  R,,H2[CCW] )                         GRA         4297     A
GRA                SX1    B6-8                                                        GRA         4298          I
          -FCC0912
GRA                SX2    B6-4                                                        GRA         4299          I
          -FCC0912
GRA                TYPEI  IS,B6,X1,X2 SRI(  IS  MI,MI-4,MI-8 )                        GRA         4300          I
          -FCC0912
GRA                BX6    X2-X5                                                       FCC0912        9     A
GRA                SX1    B6-8        RJ = MI - 8   */ OPERANDS OF *IS*               FCC0912       10     A
GRA                SX2    B6-4        RK = MI - 4                                     FCC0912       11     A
GRA       .FTN     IFEQ   HC.ID,2     IF FTN                                          CCG0071        7     A
GRA                SA4    =XHO$ARGC                                                   CCG0071        8     A
GRA                PL     X4,SIV2AC   IF NOT HO$ARGC                                  CCG0071        9     A
GRA                SB2    X1+B7                                                       CCG0071       10     A
GRA                SA4    B2+2                                                        CCG0071       11     A
GRA                LX4    59-D.FPP                                                    CCG0071       12     A
GRA                PL     X4,SIV2AA   IF NOT FP[DJ]                                   CCG0071       13     A
GRA                SX4    X1          J = RJ                                          CCG0071       14     A
GRA                SX1    B6          RJ = MI                                         CCG0071       15     A
GRA                TYPEI  SA,B6,X4    SRI(  SA  MI,J )                                CCG0071       16     A
GRA                                                                                   CCG0071       17     A
GRA       SIV2AA   SB2    X2+B7                                                       CCG0071       18     A
GRA                SA4    B2+2                                                        CCG0071       19     A
GRA                LX4    59-D.FPP                                                    CCG0071       20     A
GRA                PL     X4,SIV2C    IF NOT FP[DK]                                   CCG0071       21     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SX4    X2          J = RK                                    CCG0071      22     A
GRA                SX2    B6          RK = MI                                   CCG0071      23     A
GRA                TYPEI  SA,X2,X4    SRI(  SA  MI,J )                          CCG0071      24     A
GRA                EQ     SIV2C                                                 CCG0071      25     A
GRA                                                                              CCG0071      26     A
GRA        SIV2AC  BSS    0                                                     CCG0071      27     A
GRA        .FTN    ENDIF                                                        CCG0071      28     A
GRA        #DAL    IFNE   .DAL,0                                                FCC0912      12     A
GRA                ZR     X6,SIV2C    IF FLB = FPL2 */ NEITHER/BOTH FP&L2       FCC0912      13     A
GRA                                                                              FCC0912      14     A
GRA       *        ONE OF THE SYMBOLS IS A LEVEL2 F.P., OUTPUT A *SA* PRIOR TO  FCC0912      15     A
GRA       *        THE *IS* TO REMOVE BIT 59 IN THE APLIST WORD.                FCC0912      16     A
GRA                                                                              FCC0912      17     A
GRA                SB2    OC.SA                                                 FCC0912      18     A
GRA                NZ     X5,SIV2A    IF FPL2 " 0   */ RJ IS FP&L2              FCC0912      19     A
GRA                SX4    X1          J = RJ                                    FCC0912      20     A
GRA                SX1    B6          RJ = MI                                   FCC0912      21     A
GRA                EQ     SIV2B                                                 FCC0912      22     A
GRA        SIV2A   SX4    X2          J = RK        */ RK IS FP&L2              FCC0912      23     A
GRA                SX2    B6          RK = MI                                   FCC0912      24     A
GRA        SIV2B   TYPEI  B2,B6,X4    SRI(  SA  MI,J )                          FCC0912      25     A
GRA        #DAL    ENDIF                                                        FCC0912      26     A
GRA                                                                              FCC0912      27     A
GRA        SIV2C   TYPEI  IS,B6,X1,X2 SRI(  IS  MI,RJ,RK )                      FCC0912      28     A
GRA                ZR     X3,SIV3     IF C = 0                                  GRA        4301     A
GRA                                                                              GRA        4302     A
GRA                BX1    X3                                                    GRA        4303     A
GRA                SX7    B6-4        F = MI - 4                                GRA        4304     A
GRA                MX6    0           H2 = 0                                    GRA        4305     A
GRA                RJ     OSI         OIS(  STT  R,F,C )                        GRA        4306     A
GRA                ZR     B3,SIV      IF RF = 0                                 GRA        4307     A
GRA                                                                              GRA        4308     A
GRA        SIV3    ZR     B3,SIV4     IF RF = 0                                 GRA        4309     A
GRA                SX2    B6-4                                                  GRA        4310     A
GRA                SX3    B3                                                    GRA        4311     A
GRA                TYPEI  IA,B6,X2,X3 SRI(  IA  MI,MI-4,RF )                    GRA        4312     A
GRA                                                                              GRA        4313     A
GRA        SIV4    SX2    B6-4                                                  GRA        4314     A
GRA                TYPEI  SA,B6,X2    SRI(  SA  MI,MI-4 )                       GRA        4315     A
GRA                EQ     SIV                                                   GRA        4316     A
GRA                                                                              GRA        4317     A
GRA        SIVA    BSS    0                                                     GRA        4318          I
          -CCG1
GRA                ECHO   2,X=(RL,RB)                                           GRA        4319          I
          -CCG1
GRA                POS    P.X+L.X                                               GRA        4320          I
          -CCG1
GRA                VFD    L.X/-0                                                GRA        4321          I
          -CCG1
GRA                VFD    *P/0                                                  GRA        4322          I
          -CCG1
GRA        SIVA    BFMW   WB,(RB,CX)                                            CCG1        137          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

                -CCG16
GRA              SIVA     BFMW   WC,(RL,RB)                                  CCG16        234     A
GRA              OSI      SPACE  3,14        OSI                             GRA         4323     A
GRA             **        OSI - OUTPUT *STT* INSTRUCTION                     GRA         4324     A
GRA             *                                                            GRA         4325     A
GRA             *         ENTRY  (X1) = CAIH FIELDS                          GRA         4326     A
GRA             *                (X6,X7) = H2, RF FIELDS RIGHT JUSTIFIED     GRA         4327     A
GRA             *                                                            FCC0912       29     A
GRA             *         EXIT   (X5) = FPL2 = FP[WORDB] & LCM[WORDB]        FCC0912       30     A
GRA             *                                                            GRA         4328     A
GRA             *         USES   X - 1, 4, 5, 6, 7                           GRA         4329     A
GRA                                                                          GRA         4330     A
GRA              OSI      ROUTINE                                            GRA         4331     A
GRA                       LX7    IH.RFP                                      GRA         4332     A
GRA                       BX7    X1+X7                                       GRA         4333     A
GRA                       LX6    R1.H2P                                      GRA         4334     A
GRA                       SX4    B6                                          GRA         4335     A
GRA                       BX6    X4+X6                                       GRA         4336     A
GRA                       LX4    B1,X7                                       GRA         4337     A
GRA                       SB2    X4                                          GRA         4338     A
GRA                       SA4    A0-B2       WORDB = SYM(IH)                 GRA         4339          I
                -CCG1
GRA                       R=     X5,10B-T.LAB                                FCC0777        4          I
                -CCG1
GRA                       ERRMI  10B-T.LAB   CARRY ADDEND                    FCC0777        5          I
                -CCG1
GRA                       LX5    P.TYP                                       FCC0777        6          I
                -CCG1
GRA                       IX1    X4+X5                                       FCC0777        7          I
                -CCG1
GRA                       AX1    59                                          FCC0777        8          I
                -CCG1
GRA                       BX4    -X1*X4      IF( TYP[WORDB] \ T.LAB ) THEN WORDB = 0   FCC0777        9          I
                -CCG1
GRA                       SX5    B1                                          GRA         4340          I
                -CCG1
GRA                       LX4    -P.FPB                                      GRA         4341          I
                -CCG1
GRA                       BX1    X5*X4                                       GRA         4342          I
                -CCG1
GRA                       LX4    P.FPB-P.LCM                                 FCC0912       31          I
                -CCG1
GRA                       BX5    X1*X4       FPL2 = FP[WORDB] & LCM[WORDB]   FCC0912       32          I
                -CCG1
GRA                       SA4    F.RDT+OC.STT                                GRA         4343          I
                -CCG1
GRA                       SB2    B2+X7                                       CCG1         138     A
GRA                       SA4    A0+B2       WORDB = SYM(3*IH+1)             CCG1         139     A
GRA                       BX5    X4                                          CCG1         140     A
GRA                       LX5    59-WB.LABP                                  CCG1         141     A
GRA                       AX5    59                                          CCG1         142     A
GRA                       BX4    -X5*X4      IF( LAB[WORDB] ) THEN WORDB = 0 CCG1         143     A

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SX5    B1                                                    CCG1        144     A
GRA                LX4    -WB.FPP                                               CCG1        145     A
GRA                BX1    X5*X4                                                 CCG1        146     A
GRA                LX4    WB.FPP-WB.LCMP                                        CCG1        147     A
GRA                BX5    X1*X4       FPL2 = FP[WORDB] & LCM[WORDB]             CCG1        148     A
GRA                SA4    F.RDT+OC.STT                                          CCG1        149     A
GRA                LX1    D.FPP       FP[DB] = FP[WORDB]                        GRA        4344     A
GRA                SB2    OC.STT                                                GRA        4345     A
GRA                PX6    B2,X6                                                 GRA        4346     A
GRA                SA6    B7+B6       MOD(MI) = TYII(OC.STT,0,H2,MI)            GRA        4347     A
GRA                SA7    A6+B1       MOD(MI+1) = IHW(0,RF,CAIH)                GRA        4348          I
          -FCC1918
GRA                LX5    IH.SIAP                                               FCC1918      10     A
GRA                BX7    X5+X7                                                 FCC1918      11     A
GRA                SA7    A6+B1       MOD(MI+1) = IHW(SIA,RF,CAIH)              FCC1918      12     A
GRA                BX6    X1+X4                                                 GRA        4349     A
GRA                SA6    A7+B1       MOD(MI+2) = RDT(OC.STT) ! DB              GRA        4350     A
GRA                MX7    0                                                     GRA        4351     A
GRA                SA7    A6+B1       MOD(MI+3) = 0                             GRA        4352     A
GRA                SB6    B6+4        MI = MI + 4                               GRA        4353     A
GRA                EQ     OSI                                                   GRA        4354     A
GRA       SXC      TITLE  SXC - SET EXIT CONDITIONS ( POST STORE INFO )         GRA        4355     A
GRA       **       SXC - SET EXIT CONDITIONS ( POST STORE INFO )                GRA        4356     A
GRA       *        SETUP POST STORE INFORMATION IN *PSI* FOR LATER USE BY       GRA        4357     A
GRA       *        *IPS* IN *GPO*.                                              GRA        4358     A
GRA       *                                                                     GRA        4359     A
GRA       *        ENTRY  (NPS) = N. POST STORES IN SEQUENCE ( SET BY *CLB* )   GRA        4360     A
GRA                                                                             GRA        4361     A
GRA       SXC      ROUTINE                                                      GRA        4362     A
GRA                SA1    NPS                                                   GRA        4363     A
GRA                SA2    N.ENL                                                 GRA        4364     A
GRA                ZR     X1,SXC7     IF NPS = 0                                GRA        4365     A
GRA                AX3    B1,X2                                                 GRA        4366          I
          -FCC0985
GRA                NZ     X3,SXC1     IF N.ENL > 1                              GRA        4367          I
          -FCC0985
GRA                SA4    EPSI                                                  GRA        4368          I
          -FCC0985
GRA                NZ     X4,SXC7     IF EPSI " 0   */ *IRA* TOOK CARE OF THEM  GRA        4369          I
          -FCC0985
GRA                ZR     X2,SXC7     IF N.ENL = 0  */ LOOP WITHOUT EXITS       GRA        4370          I
          -FCC0985
GRA                                                                             GRA        4371          I
          -FCC0985
GRA       SXC1     SX0    X1+B1                                                 GRA        4372          I
          -FCC0985
GRA                SA4    EPSI                                                  FCC0985      15     A
GRA                IX3    X2-X4                                                 FCC0985      16     A
GRA                ZR     X3,SXC7     IF N.ENL = EPSI  */ NO EXIT NODE POST ST"S FCC0985      17     A
GRA                                                                             FCC0985      18     A
GRA                SX0    X1+B1                                                 FCC0985      19     A
GRA                IX1    X0*X2                                                 GRA        4373     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                ALLOC  PSI,X1+20   ALLOC( PSI , N.ENL*(NPS+1) + 20 )          GRA   4374   A
GRA                SA3    O.UDT                                                   GRA   4375   A
GRA                SA4    O.RCT                                                   GRA   4376   A
GRA                SA5    O.RXI                                                   GRA   4377   A
GRA                SB6    X2+B6       PO = O.PSI + OLD(L.PSI)                     GRA   4378   A
GRA                SB7    X2                                                      GRA   4379   A
GRA                SB3    X3+B1       (B3) = [O.UDT]+1                            GRA   4380   A
GRA                SB4    X4          (B4) = [O.RCT]                              GRA   4381   A
GRA                MX0    -IH.CAIHL                                               GRA   4382   A
GRA                SA5    X5+B1       RI = [O.RXI] + 1                            GRA   4383   A
GRA                                                                               GRA   4384   A
GRA      *         SCAN POST STORE LIST AND FORM LIST OF VARIABLES IN *PSI*      GRA   4385   A
GRA      *         THAT ARE TO BE STORED ON ENTRY TO THE LOOP EXIT BLOCKS.       GRA   4386   A
GRA                                                                               GRA   4387   A
GRA       SXC2     SA4    NPS                                                     GRA   4388   A
GRA                SB5    A5+B1       RB = RI + 1   */ LIVE ENTRY VECTOR ADDRESS  GRA   4389   A
GRA                SA7    B6          PI = PO                                     GRA   4390   A
GRA                LX5    59-E.ALLPP                                              GRA   4391   A
GRA                PL     X5,SXC5     IF ^ALLP[RI]  */ CANT POST STORE IN THIS BL GRA   4392   A
GRA                                                                               GRA   4393   A
GRA                SA0    X4          I = NPS                                     GRA   4394   A
GRA                                                                               GRA   4395   A
GRA       SXC3     SA4    PST-1+A0    J = CO[PST(I)]                              GRA   4396   A
GRA                SA3    B4+X4       CAW = RCT(J)                                GRA   4397   A
GRA                LX4    -18         REGNO = REG[PST(I)]                         GRA   4398   A
GRA                SX7    X3          K = UDI[CAW]                                GRA   4399   A
GRA                SA1    B3+X3       U2 = UDT(K+1)                               GRA   4400   A
GRA                SA2    B5+X1       LEW = [RB+WI[U2]]                           GRA   4401   A
GRA                UX6    B2,X1       BIT = BITN[U2]                              GRA   4402   A
GRA                SB2    B2-59                                                   GRA   4403   A
GRA                LX6    -B2,X2                                                  GRA   4404   A
GRA                PL     X6,SXC4     IF SHIFT(59-BIT,LEW) > 0  */ DEAD ON ENTRY  GRA   4405   A
GRA                                                                               GRA   4406   A
GRA      *         SETUP *PSI* WORD FOR THIS *RCT* ENTRY                         GRA   4407   A
GRA                                                                               GRA   4408   A
GRA                SA1    A3+B1                                                   GRA   4409   A
GRA                LX7    PS.UDIP                                                 GRA   4410   A
GRA                SX4    X4                                                      GRA   4411   A
GRA                LX4    PS.REGP                                                 GRA   4412   A
GRA                BX2    -X0*X1      CAIH = CAIH[RCT(J+1)]                       GRA   4413   A
GRA                IX6    X4+X7                                                   GRA   4414   A
GRA                BX7    X6+X2                                                   GRA   4415   A
GRA                SA7    A7+B1       PI = PI + 1;  [PI] = PSI(REGNO,K,CAIH)      GRA   4416   A
GRA                                                                               GRA   4417   A
GRA       SXC4     SB2    A0-B1                                                   GRA   4418   A
GRA                SA0    A0-B1       I = I - 1                                   GRA   4419   A
GRA                NZ     B2,SXC3     IF I > 0                                    GRA   4420   A
GRA                                                                               GRA   4421   A
GRA      *         SETUP HEADER WORD, AND POINT *BIT* ENTRY TO *PSI* LIST.       GRA   4422   A
GRA                                                                               GRA   4423   A
GRA       SXC5     SX6    A7-B6       N = PI - PO                                 GRA   4424   A
GRA                ZR     X6,SXC6     IF N = 0      */ NO STORES TO THIS BLOCK    GRA   4425   A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                                                                    GRA        4426     A
GRA                MX7    -E.BIL                                       GRA        4427     A
GRA                SA1    O.BIT                                        GRA        4428     A
GRA                LX5    1+E.ALLPP-E.BIP                              GRA        4429     A
GRA                BX4    -X7*X5      BN = BI[RI]                      GRA        4430     A
GRA                SB2    X1                                           GRA        4431     A
GRA                SA2    B2+X4       BIW = BIT(BN)                    GRA        4432     A
GRA                MX3    -BI.PIIL                                     GRA        4433     A
GRA                LX2    -BI.PIIP                                     GRA        4434     A
GRA                BX7    -X3*X2      OPI = PII[BIW]                   GRA        4435     A
GRA                ZR     X7,SXC5B    IF OPI = 0    */ NO POST ST"S FROM INNERLP  GRA  4436  A
GRA                                                                    GRA        4437     A
GRA      *         ADD POST STORES FROM INNER LOOP TO CURRENT LIST FOR THIS BLOCK  GRA  4438  A
GRA                                                                    GRA        4439     A
GRA                SB2    B7-B1                                        GRA        4440     A
GRA                SA1    B2+X7       PJ = PSI + OPI-1                 GRA        4441     A
GRA                SB2    X1          J = [PJ]      */ N.POST ST"S FROM INNER  GRA  4442  A
GRA                SX6    X6+B2       N = N + J                        GRA        4443     A
GRA                                                                    GRA        4444     A
GRA       SXC5A    SA1    A1+B1       PJ = PJ + 1                      GRA        4445     A
GRA                SB2    B2-B1       J = J - 1                        GRA        4446     A
GRA                BX7    X1                                           GRA        4447     A
GRA                SA7    A7+B1       PI = PI + 1;  [PI] = [PJ]        GRA        4448     A
GRA                NZ     B2,SXC5A    IF J " 0                         GRA        4449     A
GRA                                                                    GRA        4450     A
GRA       SXC5B    SA6    B6          [PO] = N      */ SET HEADER WORD  GRA        4451     A
GRA                BX2    X3*X2                                        GRA        4452     A
GRA                SB6    B6+B1       PO = PO + 1                      GRA        4453     A
GRA                SX4    B6-B7                                        GRA        4454     A
GRA                BX7    X2+X4       PII[BIT(BN)] = PO - PSI          GRA        4455     A
GRA                LX7    BI.PIIP                                      GRA        4456     A
GRA                SA7    A2+                                          GRA        4457     A
GRA                                                                    GRA        4458     A
GRA       SXC6     SA1    BVL                                          GRA        4459     A
GRA                SB6    B6+X6       PO = PO + N                      GRA        4460     A
GRA                SB2    X1+B1                                        GRA        4461     A
GRA                SA5    B5+B2       RI = RB + BVL+1                  GRA        4462     A
GRA                NZ     X5,SXC2     IF [RI] " 0   */ NOT END OF LIST GRA        4463     A
GRA                                                                    GRA        4464     A
GRA                SX6    B6-B7       L.PSI = PO - O.PSI               GRA        4465     A
GRA                SA6    L.PSI                                        GRA        4466     A
GRA                                                                    GRA        4467     A
GRA                TRACE  SXC,PSI                                      GRA        4468     A
GRA                                                                    GRA        4469          I
          -FCC0985
GRA       SXC      SPACE  2,10                                         FCC0985      20     A
GRA      *         ADJUST *LUV* AND *LEA* BIT VECTORS. SET BITS FOR VARIABLES  GRA  4470  A
GRA      *         EXPLICITLY REFERENCED IN THE LOOP ( *LUV* ). ADJUST THE *HB*  GRA  4471  A
GRA      *         LIVE EXIT VECTOR ( *LEA* ) TO INDICATE THAT *VC*"S THAT ARE  GRA  4472  A
GRA      *         *KD* OR RA & PL ARE DEAD ON EXIT FROM IT.           GRA        4473          I
          -FCC0392
GRA      *         *KD* ARE DEAD ON EXIT FROM IT.                      FCC0392     134     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                                                                                    GRA          4474     A
GRA       SXC7     SA5    O.RCT                                                        GRA          4475          I
         -FCC1491X
GRA       SXC7     SA1    =XOPT2                                                       FCC1491X        5          I
         -CCG4
GRA                ZR     X1,SXC      IF OPT2 = 0                                      FCC1491X        6          I
         -CCG4
GRA       SXC7     SA1    =XCO.OPT                                                     CCG4           72          I
         -CCGNC
GRA       SXC7     SA1    =XHO$OPT                                                     CCGNC         660     A
GRA                PL     X1,SXC      IF CO.OPT " 2                                    CCG4           73          I
         -CCGNC
GRA                PL     X1,SXC      IF HO$OPT " 2                                    CCGNC         661     A
GRA                                                                                    FCC1491X        7     A
GRA                SA5    O.RCT                                                        FCC1491X        8     A
GRA                SA1    LCT                                                          GRA          4476     A
GRA                SA2    O.UDT                                                        GRA          4477     A
GRA                SA3    LUV                                                          GRA          4478     A
GRA                SA4    LEA                                                          GRA          4479     A
GRA                SB4    X2+B1       (B4) = [O.UDT] + 1                               GRA          4480     A
GRA                SB2    X1                                                           GRA          4481     A
GRA                SB5    X3          (B5) = [LUV]                                     GRA          4482     A
GRA                SB6    X4          (B6) = [LEA]                                     GRA          4483     A
GRA                SA5    X5+B2       CI = O.RCT + LCT                                 GRA          4484     A
GRA                SX0    B1                                                           GRA          4485     A
GRA                NZ     B2,SXC9     IF LCT " 0    */ NO USER EXT REFS                GRA          4486     A
GRA                EQ     SXC11                                                        GRA          4487          I
         -FCC0675
GRA                EQ     SXC                                                          FCC0675       236     A
GRA                                                                                    GRA          4488     A
GRA       SXC8     SA2    B5+X4                                                        GRA          4489     A
GRA                LX1    B2,X0                                                        GRA          4490     A
GRA                BX6    X1+X2       LUV(W) = LUV(W) ! SHIFT(1,B)                     GRA          4491     A
GRA                SA6    A2                                                           GRA          4492     A
GRA                                                                                    GRA          4493     A
GRA       SXC9     SA5    A5-3        CI = CI - 3                                      GRA          4494     A
GRA                ZR     X5,SXC11    IF [CI] = 0   */ END OF TABLE                    GRA          4495          I
         -FCC0675
GRA                ZR     X5,SXC      IF [CI] = 0   */ END OF TABLE                    FCC0675       237          I
         -FCC0914
GRA                ZR     X5,SXC11    IF [CI] = 0   */ END OF TABLE                    FCC0914         5     A
GRA                SB2    X5          K = UDI[CI]                                      GRA          4496     A
GRA                ZR     B2,SXC9     IF K = 0      */ CON OR COMPILER TEMP            GRA          4497     A
GRA                                                                                    GRA          4498     A
GRA                SA4    B4+B2       U2 = UDT(K+1);  W = WI[U2]                       GRA          4499     A
GRA                LX5    59-CA.KDP                                                    GRA          4500     A
GRA                UX7    B2,X4       B = BITN[U2]                                     GRA          4501     A
GRA                MI     X5,SXC10    IF KD[CI]     */ USELESS INCREMENT               GRA          4502     A
GRA                                                                                    GRA          4503     A
GRA                LX5    CA.KDP-CA.ACP                                                GRA          4504     A
GRA                MI     X5,SXC8     IF AC[CI]                                        GRA          4505     A
GRA                LX5    CA.ACP-CA.RAP                                                GRA          4506     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                PL     X5,SXC8     IF ^RA[CI]                                    GRA        4507     A
GRA                                                                                 GRA        4508     A
GRA       *        *VC* AND *RA*, SEE IF *CORE* VALUE IS DEAD ON ENTRY TO THE LP     GRA        4509          I
          -FCC0392
GRA                                                                                 GRA        4510          I
          -FCC0392
GRA                LX5    CA.RAP-CA.PLP                                             GRA        4511          I
          -FCC0392
GRA                PL     X5,SXC9     IF ^PL[CI]    */ NOT PRELOADED IN *HB*        GRA        4512          I
          -FCC0392
GRA                LX5    CA.PLP-CA.LXP                                             GRA        4513          I
          -FCC0392
GRA                PL     X5,SXC10    IF ^LX[CI]    */ DEAD ON EXIT                 GRA        4514          I
          -FCC0392
GRA                LX5    CA.LXP-CA.PSPP                                            GRA        4515          I
          -FCC0392
GRA                PL     X5,SXC9     IF ^PSP[CI]   */ CANT POST STORE ! ^DEF       GRA        4516          I
          -FCC0392
GRA       *        *VC* AND *RA*, SET BIT IN *LUV* IF STORED IN LOOP.                FCC0392     135     A
GRA                                                                                 FCC0392     136     A
GRA                LX5    CA.RAP-CA.DEFP                                            FCC0392     137     A
GRA                PL     X5,SXC9     IF ^DEF[CI]                                   FCC0392     138     A
GRA                LX5    CA.DEFP-CA.PSPP                                           FCC0392     139     A
GRA                PL     X5,SXC8     IF ^PSP[CI]   */ STORED IN THE LOOP           FCC0392     140     A
GRA                EQ     SXC9                                                      FCC0392     141     A
GRA                                                                                 GRA        4517     A
GRA       *        CLEAR BIT IN *LEA* TO INDICATE CORE VALUE DEAD ON ENTRY TO        GRA        4518     A
GRA       *        LOOP AFTER COMBINING THE *HB* AND ITS PREDECESSOR.                GRA        4519     A
GRA                                                                                 GRA        4520     A
GRA        SXC10   SA2    B6+X4                                                     GRA        4521     A
GRA                LX1    B2,X0                                                     GRA        4522     A
GRA                BX6    -X1*X2      LEA(W) = LEA(W) & ^SHIFT(1,B)                 GRA        4523     A
GRA                SA6    A2                                                        GRA        4524     A
GRA                EQ     SXC9                                                      GRA        4525     A
GRA        SXC     SPACE  2,14                                                      FCC0914        6     A
GRA       *        ADJUST LOOP LIVE ENTRY BV TO CLEAR BITS FOR VARIABLES THAT        FCC0914        7     A
GRA       *        ARE NOT USED AND DEAD ON EXIT FROM THE LOOP.                      FCC0914        8     A
GRA                                                                                 FCC0914        9     A
GRA        SXC11   SA1    LXA                                                       FCC0914       10     A
GRA                SA2    BVL                                                       FCC0914       11     A
GRA                SA5    SVA                                                       FCC0914       12     A
GRA                SB3    X2+B1       VL = BVL + 1                                  FCC0914       13     A
GRA                SB2    B0          I = 0                                         FCC0914       14     A
GRA                SA1    X1                                                        FCC0914       15     A
GRA                SB7    X5                       (B7) = SVA                       FCC0914       16     A
GRA                                                                                 FCC0914       17     A
GRA        SXC12   SA2    B5+B2                                                     FCC0914       18     A
GRA                BX6    X1+X2       SV(I) = LX(I) ! LU(I)                         FCC0914       19     A
GRA                SA6    B7+B2                                                     FCC0914       20     A
GRA                SB2    B2+B1       I = I + 1                                     FCC0914       21     A
GRA                SA1    A1+B1                                                     FCC0914       22     A
GRA                LT     B2,B3,SXC12 IF I < VL                                     FCC0914       23     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                                                                                  FCC0914    24    A
GRA      *         CHECK FOR CHANGED BITS, AND CLEAR THOSE IN THE *LE* VECTOR        FCC0914    25    A
GRA      *         IF THE REF IS ^CM OR THE BASE MEMBER ( *CR* ) IS NOT REFED.       FCC0914    26    A
GRA                                                                                  FCC0914    27    A
GRA                SA0    B6                                                         FCC0914    28    A
GRA                SB2    B0          I = 0                                          FCC0914    29    A
GRA                SX0    3           BO = 4 - 1    */ UDT OFFSET                    FCC0914    30    A
GRA                                                                                  FCC0914    31    A
GRA       SXC13    SA2    A0+B2                     (X2) = LE(I)                     FCC0914    32    A
GRA                SA1    B7+B2                                                      FCC0914    33    A
GRA                BX3    X2*X1                                                      FCC0914    34    A
GRA                BX7    X3-X2       VD = XOR( LE(I) , SV(I) & LE(I) )              FCC0914    35    A
GRA                CX6    X7                                                         FCC0914    36    A
GRA                ZR     X6,SXC17    IF COUNT(VD) = 0                               FCC0914    37    A
GRA                                                                                  FCC0914    38    A
GRA       SXC14    RJ     FLB         J = FLB(VD)                                    FCC0914    39    A
GRA                SX1    B1                                                         FCC0914    40    A
GRA                LX1    B5,X1                                                      FCC0914    41    A
GRA                BX7    -X1*X7      VD = VD & ^SHIFT(1,J)  */ CLEAR BIT            FCC0914    42    A
GRA                SB6    B5+B5                                                      FCC0914    43    A
GRA                SX6    X0+B6       K = 2*J + BO                                   FCC0914    44    A
GRA                SA3    B4+X6       UK = UDT(K)                                    FCC0914    45    A
GRA                LX3    59-UD.CMP                                                  FCC0914    46    A
GRA                PL     X3,SXC15    IF ^CM[UK]    */ ^ A CLASS MEMBER              FCC0914    47    A
GRA                                                                                  FCC0914    48    A
GRA                LX3    1+UD.CMP-UD.BMIP                                           FCC0914    49    A
GRA                SA4    B4+X3       U2 = UDT(BMI[UK]+1)                            FCC0914    50    A
GRA                SA5    B7+X4       SVW = SV(WI[U2])                               FCC0914    51    A
GRA                UX6    B6,X4       B = BITN[U2]                                   FCC0914    52    A
GRA                SB6    B6-59                                                      FCC0914    53    A
GRA                LX6    -B6,X5                                                     FCC0914    54    A
GRA                MI     X6,SXC16    IF SHIFT(SVW,59-B) < 0  */ BASE MEM REFED      FCC0914    55    A
GRA                                                                                  FCC0914    56    A
GRA       SXC15    BX2    -X1*X2      LE(I) = LE(I) & ^SHIFT(1,J)                    FCC0914    57    A
GRA                                                                                  FCC0914    58    A
GRA       SXC16    NZ     X7,SXC14    IF VD " 0                                      FCC0914    59    A
GRA                BX6    X2                                                         FCC0914    60    A
GRA                SA6    A2                                                         FCC0914    61    A
GRA                                                                                  FCC0914    62    A
GRA       SXC17    SX0    X0+120      BO = BO + 120                                  FCC0914    63    A
GRA                SB2    B2+1        I = I + 1                                      FCC0914    64    A
GRA                LT     B2,B3,SXC13 IF I < VL                                      FCC0914    65    A
GRA                                                                                  FCC0914    66    A
GRA                EQ     SXC                                                        FCC0914    67    A
GRA       FLB      SPACE  3,14                                                       FCC0914    68    A
GRA      **        FLB - FIND LEADING BIT SET IN A WORD                              FCC0914    69    A
GRA      *                                                                           FCC0914    70    A
GRA      *         ENTRY  (X7) = WORD                                                FCC0914    71    A
GRA      *                                                                           FCC0914    72    A
GRA      *         EXIT   (B5) = BIT NUMBER ( 0 - 59 )                               FCC0914    73    A
GRA      *                                                                           FCC0914    74    A
GRA      *         USES   B - 5, 6  X - 5, 6                                         FCC0914    75    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                                                                                  FCC0914      76     A
GRA       FLB      ROUTINE                                                           FCC0914      77     A
GRA                MX5    12                                                         FCC0914      78     A
GRA                BX6    X5*X7       T12 = MASK(12) & WORD                          FCC0914      79     A
GRA                ZR     X6,FLB1     IF T12 = 0                                     FCC0914      80     A
GRA                LX6    -12                                                        FCC0914      81     A
GRA                SB5    59                                                         FCC0914      82     A
GRA                PX6    X6                                                         FCC0914      83     A
GRA                NX5    B6,X6                                                      FCC0914      84     A
GRA                SB5    B5-B6       SC = 59-NORMC(PACK(0,SHIFT(T12,-12)))          FCC0914      85     A
GRA                EQ     FLB                                                        FCC0914      86     A
GRA                                                                                  FCC0914      87     A
GRA       FLB1     PX6    X7                                                         FCC0914      88     A
GRA                SB5    47                                                         FCC0914      89     A
GRA                NX5    B6,X6                                                      FCC0914      90     A
GRA                SB5    B5-B6       SC = 47-NORMC(PACK(0,WORD))                    FCC0914      91     A
GRA                EQ     FLB                                                        FCC0914      92     A
GRA                                                                                  GRA        4526         I
          -FCC0392 -FCC0675
GRA        SXC     SPACE  1,10                                                      FCC0392     142         I
          -FCC0675
GRA       *        SCAN *TET* AND REMOVE ANY ENTRIES THAT WERE ASSIGNED TO          GRA        4527         I
          -FCC0675
GRA       *        REGISTERS IN THE LOOP.  FIRST SET THE REG FIELD FOR X-ASSGNMTS   GRA        4528         I
          -FCC0675
GRA                                                                                  GRA        4529         I
          -FCC0675
GRA        SXC11   SA1    L.TET                                                     GRA        4530         I
          -FCC0675
GRA                SA2    ITL                                                       GRA        4531         I
          -FCC0675
GRA                SB2    X1                          (B2) = L.TET                  GRA        4532         I
          -FCC0675
GRA                SB3    X2                                                        GRA        4533         I
          -FCC0675
GRA                EQ     B2,B3,SXC   IF ITL = L.TET  */ NO COMPILER TEMPS         GRA        4534         I
          -FCC0675
GRA                SA3    O.TET                                                     GRA        4535         I
          -FCC0675
GRA                SA4    NXC                                                       GRA        4536         I
          -FCC0675
GRA                SB4    X3                          (B4) = [O.TET]                GRA        4537         I
          -FCC0675
GRA                SB6    A5                          (B6) = [O.RCT]                GRA        4538         I
          -FCC0675
GRA                ZR     X4,SXC14    IF NXC = 0                                    GRA        4539         I
          -FCC0675
GRA                SX0    1           I = NXC                                       GRA        4540         I
          -FCC0675
GRA                LX0    T.REGP                                                    GRA        4541         I
          -FCC0675
GRA                                                                                  GRA        4542         I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
          -FCC0675
GRA        SXC12    SA5    XCT-1+X4    J = CO[XCT(I)]                                  GRA    4543    I
          -FCC0675
GRA                 SA3    B6+X5       CAW = RCT(J)                                    GRA    4544    I
          -FCC0675
GRA                 SB7    X3                                                          GRA    4545    I
          -FCC0675
GRA                 NZ     B7,SXC13    IF UDI[CAW] " 0                                 GRA    4546    I
          -FCC0675
GRA                 LX3    59-CA.CONP                                                  GRA    4547    I
          -FCC0675
GRA                 MI     X3,SXC13    IF CON[CAW]                                     GRA    4548    I
          -FCC0675
GRA                                                                                    GRA    4549    I
          -FCC0675
GRA                 SA1    A3+B1                                                       GRA    4550    I
          -FCC0675
GRA                 LX1    -IH.CAP     C = CA[RCT(J+1)]                                GRA    4551    I
          -FCC0675
GRA                 SA2    B4+X1                                                       GRA    4552    I
          -FCC0675
GRA                 BX6    X0+X2       REG[TET(C)] = 1  */ ANY NON-ZERO VALUE          GRA    4553    I
          -FCC0675
GRA                 SA6    A2                                                          GRA    4554    I
          -FCC0675
GRA        SXC13    SX4    X4-1        I = I - 1                                       GRA    4555    I
          -FCC0675
GRA                 NZ     X4,SXC12    IF I " 0                                        GRA    4556    I
          -FCC0675
GRA                                                                                    GRA    4557    I
          -FCC0675
GRA       *         BACKWARDS SCAN TO REMOVE ENTRIES THAT WERE ASSIGNED TO REGISTE      GRA    4558    I
          -FCC0675
GRA                                                                                    GRA    4559    I
          -FCC0675
GRA        SXC14    SB2    B2-B1       I = L.TET - 1                                   GRA    4560    I
          -FCC0675
GRA                 MX0    -T.REGL                                                     GRA    4561    I
          -FCC0675
GRA                 SB5    B2          NTL = I       */ NEW *TET* LENGTH               GRA    4562    I
          -FCC0675
GRA                 LX0    T.REGP                                                      GRA    4563    I
          -FCC0675
GRA                 SA1    =XN.GT                                                      GRA    4564    I
          -FCC0675
GRA                 SX7    0                                                           GRA    4565    I
          -FCC0675
GRA                                                                                    GRA    4566    I
          -FCC0675
GRA        SXC15    SA4    B4+B2                                                       GRA    4567    I
          -FCC0675
GRA                 BX5    -X0*X4                                                      GRA    4568    I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
          -FCC0675
GRA                ZR     X5,SXC16    IF REG[TET(I)] = 0                          GRA        4569        I
          -FCC0675
GRA                SA7    A4          TET(I) = 0                                  GRA        4570        I
          -FCC0675
GRA                LT     B2,B5,SXC16 IF I < NTL                                  GRA        4571        I
          -FCC0675
GRA                SB5    B5-B1       NTL = NTL - 1                               GRA        4572        I
          -FCC0675
GRA        SXC16   SB2    B2-B1       I = I - 1                                   GRA        4573        I
          -FCC0675
GRA                GE     B2,B3,SXC15 IF I \ ITL                                  GRA        4574        I
          -FCC0675
GRA                                                                               GRA        4575        I
          -FCC0675
GRA                SX6    B5+B1       L.TET = NTL + 1                             GRA        4576        I
          -FCC0675
GRA                IX2    X1-X6                                                   GRA        4577        I
          -FCC0675
GRA                SA6    L.TET                                                   GRA        4578        I
          -FCC0675
GRA                MI     X2,SXC      N.GT = MIN( N.GT , L.TET )                  GRA        4579        I
          -FCC0675
GRA                SA6    A1                                                      GRA        4580        I
          -FCC0675
GRA                EQ     SXC                                                     GRA        4581        I
          -FCC0675
GRA        GRA     TTL    GRA - GLOBAL REGISTER ASSIGNMENT / CODE LOOP BODY       GRA        4582     A
GRA        CLB     TITLE  CLB - CODE LOOP BODY                                    GRA        4583     A
GRA       **       CLB - CODE LOOP BODY ( CONTROL )                               GRA        4584     A
GRA                                                                               GRA        4585     A
GRA                DESCRIBE BS.,60    BSW(CL,PC,LEN)                              GRA        4586     A
GRA                DEFINE 7                                                       GRA        4587     A
GRA        CL      DEFINE 15          CODE LENGTH                                 GRA        4588     A
GRA        PC      DEFINE 2           PARCEL COUNT                                GRA        4589     A
GRA                DEFINE 18                                                      GRA        4590     A
GRA        LEN     DEFINE 18          LENGTH OF SAVED CODE IN *PIT*               GRA        4591     A
GRA                                                                               GRA        4592     A
GRA        CLPC    DEQU   PC,17       COMBINATION FIELD                           GRA        4593     A
GRA                                                                               GRA        4594     A
GRA        CLBA    BFMW   CA,(VC,PSP,LX)                                          GRA        4595          I
          -FCC0535
GRA        CLBA    BFMW   CA,(VC,PSP,LX,DEF)                                      FCC0535      12     A
GRA        CLBB    BFMW   CA,(VC,IA,RF)                                           FCC0651       5     A
GRA                                                                               GRA        4596     A
GRA        CLB     ROUTINE                                                        GRA        4597     A
GRA                SA1    NAA                                                     GRA        4598     A
GRA                SA2    NXC                                                     GRA        4599     A
GRA                IX4    X1+X2                                                   GRA        4600     A
GRA                LX3    B1,X4                                                   GRA        4601     A
GRA                SX6    X3+7+51B   NXS = 4*( 2*(NAA+NXC) + 48 ) */ EXTRA SPACE  GRA        4602     A
GRA                LX6    2                                                       GRA        4603     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SA6    NXS                                                    GRA         4604     A
GRA                                                                              GRA         4605     A
GRA      *         SCAN *RCT* AND SET *EU* BIT FOR *IA* CANIDATES WHOSE INCREMENTS GRA         4606     A
GRA      *         ARE NOT IN B-REGISTERS                                         GRA         4607          I
         -FCC0651
GRA      *         VALUE IS NOT IN A B-REGISTER AND VISA-VERSA.                   FCC0651        6     A
GRA                                                                              GRA         4608     A
GRA                MX0    -CC.REG1L                                              GRA         4609     A
GRA                SA5    LCT                                                    GRA         4610     A
GRA                SA4    O.RCT                                                  GRA         4611     A
GRA                SA1    MTAC                                                   GRA         4612          I
         -FCC0651
GRA                SA1    CLBB                                                   FCC0651        7     A
GRA                ZR     X5,CLB1     IF LCT = 0    */ NO CANIDATES              GRA         4613     A
GRA                SB6    X4                                                     GRA         4614     A
GRA                SA5    B6+X5       CI = RCT + LCT                             GRA         4615     A
GRA                SX7    B1                                                     GRA         4616     A
GRA                LX7    CA.EUP                                                 GRA         4617     A
GRA                                                                              GRA         4618     A
GRA       AUM1     SA5    A5-3        CI = CI - 3                                GRA         4619     A
GRA                ZR     X5,CLB1     IF [CI] = 0   */ END OF *RCT*              GRA         4620     A
GRA                BX6    -X1+X5                                                 GRA         4621     A
GRA                NZ     X6,AUM1     IF ^(RA,VC,IA,RF)[CI]                      GRA         4622          I
         -FCC0651
GRA                NZ     X6,AUM1     IF ^(VC,IA,RF)[CI]                         FCC0651        8     A
GRA                SA4    A5+2        CCW = CI + 2                               GRA         4623     A
GRA                LX4    -CC.INCP                                               GRA         4624          I
         -FCC1626
GRA                LX4    -CC.REG1P                                              FCC1626        1     A
GRA                BX6    -X0*X4                                                 FCC1626        2     A
GRA                LX4    CC.REG1P-CC.INCP                                       FCC1626        3     A
GRA                AX6    3                                                      FCC1626        4     A
GRA                SA3    B6+X4       CJ = RCT(INC[CCW])                         GRA         4625     A
GRA                LX3    59-CA.RAP                                              GRA         4626     A
GRA                ZR     X3,AUM1     IF [CJ] = 0   */ INC[CCW] = 0              GRA         4627     A
GRA                LX5    59-CA.RAP                                              FCC0651        9     A
GRA                PL     X5,AUM3     IF ^RA[CI]    */ INC VAR NOT IN A B-REG    FCC0651       10     A
GRA                LX5    1+CA.RAP                                               FCC0651       11     A
GRA                NZ     X6,AUM3     IF REG1[CCW] \ 20B  */ INC VAR IN AN X-REG FCC1626        5     A
GRA                LX4    CC.INCP+59-CC.I2P                                      GRA         4628     A
GRA                MI     X4,AUM2     IF I2[CCW]                                 GRA         4629     A
GRA                PL     X3,AUM2     IF ^RA[CJ]                                 GRA         4630     A
GRA                SA3    A3+2        CCWJ = CJ + 2                              GRA         4631     A
GRA                LX3    -CC.REG1P                                              GRA         4632     A
GRA                BX6    -X0*X3                                                 GRA         4633     A
GRA                AX6    3                                                      GRA         4634     A
GRA                ZR     X6,AUM1     IF REG1[CCWJ] < 7  */ INC IN A B-REG       GRA         4635     A
GRA                                                                              GRA         4636     A
GRA       AUM2     BX6    X7+X5       EU[CI] = 1                                 GRA         4637     A
GRA                SA6    A5                                                     GRA         4638     A
GRA                EQ     AUM1                                                   GRA         4639     A
GRA                                                                              FCC0651       12     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA       AUM3     LX3    1+CA.RAP                                                FCC0651      13     A
GRA                BX6    X7+X3       EU[CJ] = 1    */ FORCE INC VAL TO BE *EU*   FCC0651      14     A
GRA                SA6    A3                                                      FCC0651      15     A
GRA                EQ     AUM1                                                    FCC0651      16     A
GRA       CLB      SPACE  2,10                                                    GRA        4640     A
GRA      *         INITIALIZE FLAGS                                               GRA        4641     A
GRA                                                                               GRA        4642     A
GRA       CLB1     SX6    B1+B1                                                   GRA        4643     A
GRA                MX7    0                                                       GRA        4644     A
GRA                SA6    BSI         BSI = 2                                     GRA        4645     A
GRA                SA7    LBN         LBN = 0       */ LAST USE BLOCK NUMBER      GRA        4646          I
         -FCC0675
GRA                SA7    MRU         MRU = 0       */ MACHINE REGS USED          GRA        4647     A
GRA                SA7    L.PIT       L.PIT = 0                                   GRA        4648     A
GRA                SA7    PL          PL = 0                                      GRA        4649     A
GRA                                                                               GRA        4650     A
GRA      *         SCAN *RAT* AND FORM POST STORE LIST                            GRA        4651     A
GRA                                                                               GRA        4652     A
GRA                SX2    27B         I = 27B                                     GRA        4653     A
GRA                SA5    O.RCT                                                   GRA        4654     A
GRA                SA3    NXC                                                     GRA        4655     A
GRA                SA1    CLBA                                                    GRA        4656     A
GRA                SB7    B0          NPS = 0                                     GRA        4657     A
GRA                SB6    X5                                                      GRA        4658     A
GRA                NZ     X3,CLB2     IF NXC " 0                                  GRA        4659     A
GRA                SX2    7           I = 7                                       GRA        4660     A
GRA                                                                               GRA        4661     A
GRA       CLB2     SA5    RAT+X2      J = R1[RAT(I)]                              GRA        4662     A
GRA                ZR     X5,CLB3     IF RAT(I) = 0                               GRA        4663     A
GRA                SA4    B6+X5       CAW = RCT(J)                                GRA        4664     A
GRA                BX6    -X1+X4                                                  GRA        4665     A
GRA                NZ     X6,CLB3     IF ^( VC[CAW] & PSP[CAW] & LX[CAW] )        GRA        4666          I
         -FCC0535
GRA                NZ     X6,CLB3     IF ^(VC[CAW]&PSP[CAW]&LX[CAW]&DEF[CAW])     FCC0535      13     A
GRA                                                                               GRA        4667     A
GRA                SX3    X5                                                      GRA        4668     A
GRA                LX2    18                                                      GRA        4669     A
GRA                BX7    X2+X3                                                   GRA        4670     A
GRA                LX2    -18                                                     GRA        4671     A
GRA                SA7    PST+B7      PST(NPS) = SHIFT(I,18) ! J                  GRA        4672     A
GRA                SB7    B7+1        NPS = NPS + 1                               GRA        4673     A
GRA                                                                               GRA        4674     A
GRA       CLB3     SX2    X2-1        I = I - 1                                   GRA        4675     A
GRA                NZ     X2,CLB2     IF I " 0                                    GRA        4676     A
GRA                                                                               GRA        4677     A
GRA                SX7    B7+                                                     GRA        4678     A
GRA                SA7    NPS                                                     GRA        4679     A
GRA                                                                               GRA        4680     A
GRA      *         PROCESS ALL NON-EMPTY BLOCKS IN THE LOOP BODY                  GRA        4681     A
GRA                                                                               GRA        4682     A
GRA       CLB4     SX6    4           L.TXT = 4                                   GRA        4683     A
GRA                SA6    L.TXT                                                   GRA        4684     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                CALL   RTB         READ BLOCK IN                                  GRA         4685          I
          -CCGNC
GRA                CALL   RTB#        READ BLOCK IN                                  CCGNC        662     A
GRA                SA1    L.SEQ                                                      GRA         4686     A
GRA                SB2    X1-8                                                       GRA         4687          I
          -FCC2051
GRA                LE     B2,CLB10    IF L.SEQ @ 8  */ EMPTY BLOCK                   GRA         4688          I
          -FCC2051
GRA                                                                                  GRA         4689          I
          -FCC2051
GRA                RJ     SLW         SAVE LINK WORDS                                GRA         4690     A
GRA                                                                                  GRA         4691     A
GRA                CALL   IRA         INSERT REGISTER ASSIGNMENTS                    GRA         4692     A
GRA                                                                                  FCC2051        9     A
GRA                SA1    L.TXT                                                      FCC2051       10     A
GRA                SA2    DTL                                                        FCC2051       11     A
GRA                IX3    X1-X2                                                      FCC2051       12     A
GRA                ZR     X3,CLB10    IF L.TXT = DTL  */ NOTHING TO CODE             FCC2051       13          I
          -FCCA509X
GRA                NZ     X3,CLB4A    IF L.TXT " DTL */CODE IT                       FCCA509X      26     A
GRA                                                                                  FCCA509X      27     A
GRA       *        NOTHING TO CODE - MAKE BST(BSI+1) = 0                             FCCA509X      28     A
GRA       *        SO NOTHING WILL BE WRITTEN TO BLK.                                FCCA509X      29     A
GRA                                                                                  FCCA509X      30     A
GRA                SA1    O.BST                                                      FCCA509X      31     A
GRA                SA2    BSI                                                        FCCA509X      32     A
GRA                MX6    0                                                          FCCA509X      33     A
GRA                IX3    X1+X2                                                      FCCA509X      34     A
GRA                SA6    X3+B1       BST(BSI+1) = 0                                 FCCA509X      35     A
GRA                EQ     CLB10                                                      FCCA509X      36     A
GRA                                                                                  FCCA509X      37     A
GRA       CLB4A    BSS    0                                                          FCCA509X      38     A
GRA                                                                                  FCC2051       14     A
GRA                TRACE  CLB,RLIST,SEQ                                              GRA         4693     A
GRA                TRACE  CLB,RLIST                                                  GRA         4694     A
GRA                                                                                  GRA         4695     A
GRA                RJ     RLW         RESTORE LINK WORDS                             GRA         4696     A
GRA                                                                                  GRA         4697     A
GRA                CALL   CUC         COLLECT USES COUNTS                            GRA         4698     A
GRA                                                                                  GRA         4699     A
GRA       #IWT     IFEQ   .CPU,74                                                    GRA         4700     A
GRA       #IWT     IFNE   .IWT,0                                          ?IWT       GRA         4701     A
GRA                                                                                  GRA         4702     A
GRA       *        IF A 6600 AND IWTIME " 0, THEN CHECK LOOP LENGTH AND SET STACK    GRA         4703     A
GRA       *        FLAG IF IT FITS.  NOTE THAT BECAUSE OF PARCEL FAULTS THE          GRA         4704     A
GRA       *        LOOP COULD BE CODED BY *MCG* AND NOT FIT *INSTACK*.               GRA         4705     A
GRA                                                                                  GRA         4706     A
GRA                SA1    N.EBB                                                      GRA         4707     A
GRA                SX6    X1-1                                                       GRA         4708     A
GRA                NZ     X6,CLS3     IF N.EBB " 1                                   GRA         4709     A
GRA                SA5    O.TXT                                                      GRA         4710     A
GRA                MX1    -D.SZL                                                     GRA         4711     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SA4    L.TXT                                                GRA        4712    A
GRA                LX1    D.SZP                                                GRA        4713    A
GRA                SX2    B1+B1                                                GRA        4714    A
GRA                SX7    29                                                   GRA        4715    A
GRA                LX2    D.SZP       LEN = 2                                  GRA        4716    A
GRA                IX6    X4+X5                                                GRA        4717    A
GRA                LX7    D.SZP                                                GRA        4718    A
GRA                SA4    X6-2        DI = O.TXT + L.TXT - 2                   GRA        4719    A
GRA                SB4    4                                                    GRA        4720    A
GRA                                                                            GRA        4721    A
GRA        CLS1    SA4    A4-B4       DI = DI - 4                              GRA        4722    A
GRA                LX4    59-D.JPP                                             GRA        4723    A
GRA                PL     X4,CLS1     IF ^JP[DI]    */ NOT TERMINAL JUMP       GRA        4724    A
GRA                                                                            GRA        4725    A
GRA        CLS2    SA4    A4-B4       DI = DI - 4                              GRA        4726    A
GRA                SA5    A4-2        R1 = DI - 2                              GRA        4727    A
GRA                BX6    -X1*X4                                               GRA        4728    A
GRA                IX2    X2+X6       LEN = LEN + SZ[DI]                       GRA        4729    A
GRA                UX3    B2,X5                                                GRA        4730    A
GRA                IX6    X2-X7                                                GRA        4731    A
GRA                PL     X6,CLS3     IF LEN > 28   */ LOOP NOT INSTACK        GRA        4732    A
GRA                NE     B2,B1,CLS2  IF OC[R1] " OC.BOS                       GRA        4733    A
GRA                                                                            GRA        4734    A
GRA                MX7    1                                                    GRA        4735    A
GRA                LX7    1+IH.STKP                                            GRA        4736    A
GRA                SA7    A5+5        STK[R1+5] = 1  */ INDICATE STACK TIMING  GRA        4737    A
GRA                                                                            GRA        4738    A
GRA        CLS3    BSS    0                                                    GRA        4739    A
GRA        #IWT    ENDIF                                              'IWT      GRA        4740    A
GRA                                                                            GRA        4741    A
GRA                SA3    NXC                                                  GRA        4742    A
GRA                RJ     CIS         SET THE PARCEL COUNT AND CODE THE SEQUENCE GRA      4743    A
GRA                NZ     X6,CLB14    IF MCGFAIL " 0  */ TOO MANY LOCKED REGS  GRA        4744    A
GRA                                                                            GRA        4745    A
GRA        *       COLLECT THE LIST OF REGISTERS USED BY *MCG* FOR OUTER LOOPS GRA        4746    A
GRA                                                                            GRA        4747    A
GRA                SA5    MRA                                                  GRA        4748    A
GRA                ZR     X5,CLB9     IF MRA = 0    */ NO REGS AVAIL           GRA        4749    A
GRA                                                                            GRA        4750    A
GRA                SA1    MRU                                                  GRA        4751    A
GRA                SA2    O.TXT                                                GRA        4752    A
GRA                SA3    L.TXT                                                GRA        4753    A
GRA                SA5    X2+7        LI = O.TXT + 7                           GRA        4754    A
GRA                SB4    4                                                    GRA        4755    A
GRA                SB3    X3-8        I = L.TXT - 8                            GRA        4756    A
GRA                SX0    B1                                                   GRA        4757    A
GRA                BX6    X1                                                   GRA        4758    A
GRA                                                                            GRA        4759    A
GRA        CLB8    SB5    X5          REGN = REG[LI]                           GRA        4760    A
GRA                LX2    B5,X0                                                GRA        4761    A
GRA                SB3    B3-B4       I = I - 4                                GRA        4762    A
GRA                BX6    X2+X6       MRU = MRU ! SHIFT(1,REGN)                GRA        4763    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SA5    A5+B4       LI = LI + 4                                GRA        4764     A
GRA                NZ     B3,CLB8     IF I " 0                                   GRA        4765     A
GRA                SA6    A1                                                     GRA        4766     A
GRA                                                                              GRA        4767     A
GRA      *         SAVE CODE LENGTH INFO IN *BST*                                GRA        4768     A
GRA                                                                              GRA        4769     A
GRA       CLB9     SA1    PL                                                     GRA        4770     A
GRA                SA2    L.PIT                                                  GRA        4771     A
GRA                SA4    =XPC                                                   GRA        4772     A
GRA                SX3    0           CODEL = 0                                  GRA        4773     A
GRA                SA5    O.BST                                                  GRA        4774     A
GRA                IX0    X2-X1       LEN = L.PIT - PL                           GRA        4775     A
GRA                LX0    BS.LENP                                                GRA        4776     A
GRA                BX6    X2          PL = L.PIT                                 GRA        4777     A
GRA                SA6    A1                                                     GRA        4778     A
GRA                LX3    BS.CLP                                                 GRA        4779     A
GRA                LX4    BS.PCP                                                 GRA        4780     A
GRA                BX3    X3+X4                                                  GRA        4781     A
GRA                SA4    BSI                                                    GRA        4782     A
GRA                IX6    X3+X0                                                  GRA        4783     A
GRA                SB5    X5+B1                                                  GRA        4784     A
GRA                SA6    B5+X4       BST(BSI+1) = BSW(CL,PC,LEN)                GRA        4785     A
GRA                MX7    0                                                      GRA        4786     A
GRA                SA1    NXC                                                    GRA        4787     A
GRA                NZ     X1,CLB10    IF NXC " 0    */ LOCKED X-MODE             GRA        4788     A
GRA                SA1    NAA                                                    FCC0603      40     A
GRA                NZ     X1,CLB10    IF NAA " 0    */ LOCKED A-REGS             FCC0603      41     A
GRA                                                                              GRA        4789     A
GRA                SA7    PL          PL = 0                                     GRA        4790     A
GRA                BX4    X6                                                     GRA        4791     A
GRA                SA1    A6-B1       BAW = BST(BSI)                             GRA        4792     A
GRA                RJ     WSC         DUMP CODE                                  GRA        4793     A
GRA                SA7    PL          PL = 0                                     GRA        4794     A
GRA                SA7    L.PIT       L.PIT = 0                                  GRA        4795     A
GRA                                                                              GRA        4796     A
GRA      *         ADVANCE TO NEXT SEQUENCE                                      GRA        4797     A
GRA                                                                              GRA        4798     A
GRA       CLB10    SA1    BSI                                                    GRA        4799     A
GRA                SA2    L.BST                                                  GRA        4800     A
GRA                SX6    X1+2        BSI = BSI + 2                              GRA        4801     A
GRA                SX3    X2-1                                                   GRA        4802     A
GRA                IX7    X6-X3                                                  GRA        4803     A
GRA                SA6    A1                                                     GRA        4804     A
GRA                MI     X7,CLB4     IF BSI < L.BST - 1                         GRA        4805     A
GRA                                                                              GRA        4806     A
GRA                SX7    4                                                      GRA        4807     A
GRA                SA7    L.TXT       L.TXT = 4                                  GRA        4808     A
GRA       CLB      SPACE  1,10                                                   GRA        4809     A
GRA      *         WRITE THE SAVED CODE BACK TO *BLK*                            GRA        4810     A
GRA                                                                              GRA        4811     A
GRA                SA1    NXC                                                    GRA        4812          I
          -FCC0603

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                ZR     X1,CLB      IF NXC = 0    */ NO SAVED CODE                GRA         4813          I
          -FCC0603
GRA                SA1    L.PIT                                                     FCC0603       42     A
GRA                ZR     X1,CLB      IF L.PIT = 0  */ NO SAVED CODE                FCC0603       43     A
GRA                SX6    B1+B1       BSI = 2                                       GRA         4814     A
GRA                MX7    0                                                         GRA         4815     A
GRA                SA7    PL          PL = 0                                        GRA         4816     A
GRA                                                                                GRA         4817     A
GRA       CLB11    SA2    O.BST                                                     GRA         4818     A
GRA                IX7    X6+X2                                                     GRA         4819     A
GRA                SA1    X7          BAW = BST(BSI)                                GRA         4820     A
GRA                SA4    X7+B1       BSW = BST(BSI+1)                              GRA         4821     A
GRA                ZR     X1,CLB13    IF BAW = 0    */ END OF *BST*                 GRA         4822     A
GRA                LX1    -BA.LENP                                                  GRA         4823          I
          -FCCA509X
GRA                SB2    X1-8                                                      GRA         4824          I
          -FCCA509X
GRA                LE     B2,CLB12    IF LEN[BAW] @ 8  */ EMPTY BLOCK               GRA         4825          I
          -FCCA509X
GRA                SA6    BSI                                                       GRA         4826          I
          -FCCA509X
GRA                LX1    BA.LENP                                                   GRA         4827          I
          -FCCA509X
GRA                ZR     X4,CLB12    IF BST(BSI+1) = 0 */NOT CODED                 FCCA509X      39     A
GRA                SA6    BSI                                                       FCCA509X      40     A
GRA                RJ     WSC         WRITE SAVED CODE TO *BLK*                     GRA         4828     A
GRA                SA1    BSI                                                       GRA         4829     A
GRA                SX6    X1+                                                       GRA         4830     A
GRA       CLB12    SX6    X6+2        BSI = BSI + 2                                 GRA         4831     A
GRA                EQ     CLB11                                                     GRA         4832     A
GRA                                                                                GRA         4833     A
GRA       CLB13    SX6    0                                                         GRA         4834     A
GRA                SA6    L.PIT       L.PIT = 0                                     GRA         4835     A
GRA                EQ     CLB                                                       GRA         4836     A
GRA       CLB      SPACE  3,10                                                      GRA         4837     A
GRA      *         *MCG* FAILED, REDUCE LOCKED X-REG COUNT AND TRY AGAIN.           GRA         4838     A
GRA                                                                                GRA         4839     A
GRA       CLB14    SA1    NXC         NXC = NXC - 1                                 GRA         4840          I
          -FCC0811
GRA                SA3    O.RCT                                                     GRA         4841          I
          -FCC0811
GRA                MX0    1                                                         GRA         4842          I
          -FCC0603 -FCC0811
GRA                SA2    XCT-1+X1    J = CO[XCT(NXC)]                              GRA         4843          I
          -FCC0603 -FCC0811
GRA                ZR     X1,CLB17    IF NXC < 0                                    FCC0603       44          I
          -FCC0811
GRA                SA2    XCT-1+X1    J = CO[XCT(NXC)]                              FCC0603       45          I
          -FCC0811
GRA                MX0    1                                                         FCC0603       46          I
          -FCC0811
GRA                SB3    X3                                                        GRA         4844          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
          -FCC0811
GRA                 LX0    1+CA.RAP                                    GRA        4845        I
          -FCC0811
GRA                 SA4    B3+X2                                       GRA        4846        I
          -FCC0811
GRA                 BX6    -X0*X4      RA[RCT(J)] = 0                  GRA        4847        I
          -FCC0811
GRA                 SA5    A4+2                                        GRA        4848        I
          -FCC0811
GRA                 SX7    X1-1                                        GRA        4849        I
          -FCC0811
GRA                 SA7    A1                                          GRA        4850        I
          -FCC0811
GRA                 MX0    -CC.REG1L                                   GRA        4851        I
          -FCC0811
GRA                 SA6    A4                                          GRA        4852        I
          -FCC0811
GRA                 LX5    -CC.REG1P                                   GRA        4853        I
          -FCC0811
GRA                 BX1    -X0*X5      REGN = REG1[RCT(J+2)]           GRA        4854        I
          -FCC0811
GRA                 MX7    0                                           GRA        4855        I
          -FCC0811
GRA                 SA7    RAT+X1      RAT(REGN) = 0                   GRA        4856        I
          -FCC0811
GRA                 SA7    RVT+X1      RVT(REGN) = 0                   GRA        4857        I
          -FCC0811
GRA       CLB14     SA1    NXC                                         FCC0811       9     A
GRA                 ZR     X1,CLB17    IF NXC = 0                      FCC0811      10     A
GRA                 RJ     RXA         REMOVE LAST X-ASSIGNMENT        FCC0811      11     A
GRA                 PRINT  CLB,(* --- MCG FAIL ---*)                   GRA        4858     A
GRA                 EQ     CLB1                                        GRA        4859     A
GRA       CLB       SPACE  2,10                                        FCC0603      47     A
GRA      *          A-REGS LOCKED AND MCG FAILED, REDUCE *MARA* AND RESTART  FCC0603  48  A
GRA                                                                    FCC0603      49     A
GRA       CLB17     CLCM   CLW         CLEAR LINK WORDS                FCC0603      50     A
GRA                 SA1    NAA                                         FCC0603      51     A
GRA                 SX6    X1-1        MARA = NAA - 1                  FCC0603      52     A
GRA                 SA6    MARA                                        FCC0603      53     A
GRA                 SX7    4                                           FCC0603      54     A
GRA                 SA7    L.TXT       L.TXT = 4                       FCC0603      55     A
GRA                 SX7    0                                           FCC0603      56     A
GRA                 SA7    L.PIT       L.PIT = 0                       FCC0603      57     A
GRA                 SA7    L.RCT       L.RCT = 0                       FCC0603      58     A
GRA                 SA7    LUL+1       LUL(2) = 0                      FCC0603      59     A
GRA                 MX6    36                                          FCC0603      60     A
GRA                 BX6    -X6                                         FCC0603      61     A
GRA                 SA6    MRA         MRA = ^MASK(36)                 FCC0603      62     A
GRA                 EQ     GRA0                                        FCC0603      63     A
GRA       CLW       SPACE  3                                           FCC0603      64     A
GRA      **         CLW - CLEAR LINK WORDS OF A BLOCK                  FCC0603      65     A
GRA                                                                    FCC0603      66     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA        CLW      ROUTINE                                                  FCC0603      67     A
GRA                 SA1    O.SEQ                                             FCC0603      68     A
GRA                 SA2    L.SEQ                                             FCC0603      69     A
GRA                 SB4    4                                                 FCC0603      70     A
GRA                 SB2    X1+3        I = 3                                 FCC0603      71     A
GRA                 IX3    X1+X2                                             FCC0603      72     A
GRA                 MX6    0                                                 FCC0603      73     A
GRA                 SB3    X3                                                FCC0603      74     A
GRA                                                                          FCC0603      75     A
GRA        CLW1     SA6    B2          SEQ(I) = 0                            FCC0603      76     A
GRA                 SB2    B2+B4       I = I + 4                             FCC0603      77     A
GRA                 LE     B2,B3,CLW1  IF I < L.SEQ                          FCC0603      78     A
GRA                                                                          FCC0603      79     A
GRA                 EQ     CLW                                               FCC0603      80     A
GRA        RXA      SPACE  3,14                                              FCC0811      12     A
GRA       **        RXA - REMOVE AN X-ASSIGNMENT FROM *RCT*                  FCC0811      13     A
GRA       *                                                                  FCC0811      14     A
GRA       *         ENTRY  (A1,X1) = NXC,[NXC]                               FCC0811      15     A
GRA       *                                                                  FCC0811      16     A
GRA       *         EXIT   (X6) = NXC , DECREMENTED                          FCC0811      17     A
GRA                                                                          FCC0811      18     A
GRA        RXA      ROUTINE                                                  FCC0811      19     A
GRA                 SA3    O.RCT                                             FCC0811      20     A
GRA                 SA2    XCT-1+X1    J = CO[XCT(NXC)]                      FCC0811      21     A
GRA                 MX0    2                                                 FCC0811      22     A
GRA                 LX0    2+CA.RAFP                                         FCC0811      23     A
GRA                 SB3    X3                                                FCC0811      24     A
GRA                 SA4    B3+X2                                             FCC0811      25     A
GRA                 BX6    -X0*X4      (RA,IRA)RCT(J) = 0                    FCC0811      26     A
GRA                 SA6    A4                                                FCC0811      27     A
GRA                 MX0    -CC.REG1L                                         FCC0811      28     A
GRA                 SA5    A4+2        CCW = RCT(J+2)                        FCC0811      29     A
GRA                 LX5    -CC.REG1P                                         FCC0811      30     A
GRA                 BX2    -X0*X5      REGN = REG1[CCW]                      FCC0811      31     A
GRA                 MX0    -CC.REGFL                                         FCC0811      32     A
GRA                 LX0    CC.REGFP                                          FCC0811      33     A
GRA                 LX5    CC.REG1P                                          FCC0811      34     A
GRA                 BX6    X0*X5       REGF[RCT(J+2)] = 0                    FCC0811      35     A
GRA                 SA6    A5                                                FCC0811      36     A
GRA                 SX6    X1-1        NXC = NXC - 1                         FCC0811      37     A
GRA                 SA6    A1                                                FCC0811      38     A
GRA                 ZR     X2,RXA      IF REGN = 0                           FCC0811      39     A
GRA                 MX7    0                                                 FCC0811      40     A
GRA                 SA7    RAT+X2      RAT(REGN) = 0                         FCC0811      41     A
GRA                 SA7    RVT+X2      RVT(REGN) = 0                         FCC0811      42     A
GRA                 EQ     RXA                                               FCC0811      43     A
GRA        CXB      TITLE  CXB - CODE EXTENDED BASIC BLOCK                   GRA         4860     A
GRA       **        CXB - CODE EXTENDED BASIC BLOCK ( STRAIGHT LINE CODE )   GRA         4861     A
GRA                                                                          GRA         4862     A
GRA        CXB      ENTRY. **                                                GRA         4863          I
          -CCGNC
GRA        CXB      ENTRY. **,#                                              CCGNC        663     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SA1    L.SEQ                                                GRA        4864     A
GRA                SB2    X1-8                                                 GRA        4865     A
GRA                LE     B2,CXB      IF L.SEQ @ 8                             GRA        4866     A
GRA                                                                            GRA        4867     A
GRA                CALL   AIS         ASSIGN INDEX REGISTERS                   GRA        4868          I
          -CCGNC
GRA                CALL   AIS#        ASSIGN INDEX REGISTERS                   CCGNC       664     A
GRA                MX6    0                                                    GRA        4869     A
GRA                SA6    PL          PL = 0                                   GRA        4870     A
GRA                MX3    0                                                    GRA        4871     A
GRA                RJ     CIS         SET *PC* AND CODE SEQUENCE               GRA        4872     A
GRA                SA1    O.BST                                                GRA        4873     A
GRA                SA2    BSI                                                  GRA        4874     A
GRA                SA3    =XPC                                                 GRA        4875     A
GRA                SA4    L.PIT                                                GRA        4876     A
GRA                IX7    X1+X2                                                GRA        4877     A
GRA                SA1    X7          BAW = BST(BSI)                           GRA        4878     A
GRA                LX3    BS.PCP                                               GRA        4879     A
GRA                BX4    X3+X4       BSW = BSW(0,PC,L.PIT)                    GRA        4880     A
GRA                RJ     WSC         WRITE SAVED CODE TO *BLK*                GRA        4881     A
GRA                SX6    4                                                    GRA        4882     A
GRA                SA6    L.TXT       L.TXT = 4                                GRA        4883     A
GRA                SA7    L.PIT       L.PIT = 0                                GRA        4884     A
GRA                EQ     CXB                                                  GRA        4885     A
GRA         CIS    SPACE  2,14        CIS                                      GRA        4886     A
GRA        **      CIS - CODE INSTRUCTION SEQUENCE                             GRA        4887     A
GRA        *               SETS THE INITIAL VALUE OF *PARCEL* AND CALLS *MCG*. GRA        4888     A
GRA        *                                                                   GRA        4889     A
GRA        *       ENTRY  (X3) = NXC , *MCG* LOCKED REGISTER FLAG              GRA        4890     A
GRA        *       EXIT   (X6) = MCGFAIL FLAG, = 0 IF SEQUENCE CODED           GRA        4891     A
GRA                                                                            GRA        4892     A
GRA         CIS    ROUTINE                                                     GRA        4893     A
GRA                SA1    O.BST                                                GRA        4894     A
GRA                SA2    O.BIT                                                GRA        4895     A
GRA                SA4    BSI         J = BSI                                  GRA        4896     A
GRA                SB2    X1                                                   GRA        4897     A
GRA                SB3    X2                                                   GRA        4898     A
GRA                SA4    B2+X4                                                GRA        4899     A
GRA                LX4    -BA.BIP     BI = BI[BST(J)]                          GRA        4900     A
GRA                SA5    B3+X4       BIW = BIT(BI)                            GRA        4901     A
GRA                MX7    0           PPC = 0       */ PREVIOUS SEQ *PC*       GRA        4902     A
GRA                LX5    59-BI.ILP                                            GRA        4903     A
GRA                MI     X5,CIS2     IF IL[BIW]    */ BLOCK HAS A INITIAL LABEL GRA        4904     A
GRA                MX6    -BI.PCL                                              GRA        4905     A
GRA                                                                            GRA        4906     A
GRA         CIS1   SA5    A5-2        BI = BI - 2;  BIW = BIT(BI)              GRA        4907     A
GRA                LX5    59-BI.CBP                                            GRA        4908     A
GRA                PL     X5,CIS1     IF ^CB[BIW]   */ PART OF AN EXTENDED BLOCK GRA        4909     A
GRA                LX5    1+BI.CBP-BI.PCP                                      GRA        4910     A
GRA                BX7    -X6*X5      PPC = PC[BIW]                            GRA        4911     A
GRA                                                                            GRA        4912     A
GRA         CIS2   SX6    X3+                                                  GRA        4913     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                 SA7    =XPARCEL    PARCEL = PPC                                  GRA          4914          I
            -CCG7
GRA                 SA7    =XCG.PC     CG.PC = PPC                                   CCG7          115          I
            -CCGNC
GRA                 SA7    =XCC$PC     CC$PC = PPC                                   CCGNC         665     A
GRA                 CALL   MCG (NXC )  CODE THE SEQUENCE                             GRA          4915          I
            -CCGNC
GRA                 CALL   MCG# ( NXC )  CODE THE SEQUENCE                           CCGNC         666     A
GRA                 EQ     CIS                                                       GRA          4916     A
GRA          WSC    TITLE  WSC - WRITE SAVED CODE FROM *PIT* TO *BLK*                GRA          4917     A
GRA        **       WSC - WRITE SAVED CODE IN *PIT* TO *BLK*                         GRA          4918     A
GRA        *                                                                         GRA          4919     A
GRA        *        ENTRY  (BSI) = *BST* INDEX                                       GRA          4920     A
GRA        *               (PL) = INDEX TO SAVED CODE IN *PIT*                       GRA          4921     A
GRA        *               (X1) = BAW = BST(BSI)                                     GRA          4922     A
GRA        *               (X4) = BSW = BST(BSI+1) = BSW(CL,PC,LEN)                  GRA          4923     A
GRA        *                                                                         GRA          4924     A
GRA        *        EXIT   (M.PL) = MAX *PIT* LENGTH OF A BLOCK UPDATED              GRA          4925     A
GRA        *               (LBN) = LAST BLOCK NUMBER OF REGION CODED UPDATED         GRA          4926          I
            -FCC0675
GRA        *               (X7) = 0                                                  GRA          4927     A
GRA        *               *CB* AND *PC* FIELDS SET IN *BIT* ENTRY.                  GRA          4928     A
GRA                                                                                  GRA          4929     A
GRA          WSC    ROUTINE                                                           GRA          4930     A
GRA                 SA5    LBN                                                       GRA          4931          I
            -FCC0675
GRA                 SA3    O.BIT                                                     GRA          4932          I
            -FCC0675
GRA                 LX1    -BA.BIP                                                   GRA          4933          I
            -FCC0675
GRA                 MX0    -BS.CLPCL                                                 GRA          4934          I
            -FCC0675
GRA                 SB3    X3                        (B3) = [O.BIT]                  GRA          4935          I
            -FCC0675
GRA                 LX0    BS.CLPCP                                                  GRA          4936          I
            -FCC0675
GRA                 SX6    X1          BN = BI[BAW]                                  GRA          4937          I
            -FCC0675
GRA                 IX7    X5-X6                                                     GRA          4938          I
            -FCC0675
GRA                 PL     X7,WSC1     IF LBN \ BN                                   GRA          4939          I
            -FCC0675
GRA                 SA1    NPB                                                       GRA          4940          I
            -FCC0675
GRA                 LX1    1                                                         GRA          4941          I
            -FCC0675
GRA                 IX7    X6-X1                                                     GRA          4942          I
            -FCC0675
GRA                 SB2    X7                                                        GRA          4943          I
            -FCC0675
GRA                 GT     B2,WSC1     IF BN > 2*NPB */ HOLDING BLOCK                GRA          4944          I
            -FCC0675

            0         1         2         3         4         5         6         7         8
            12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                 SA6    A5          LBN = BN                                  GRA        4945          I
          -FCC0675
GRA                                                                              GRA        4946          I
          -FCC0675
GRA        WSC1     SA2    B3+X6       BIW = BIT(BN)                             GRA        4947          I
          -FCC0675
GRA                 SA3    M.PL                                                  GRA        4948          I
          -FCC0675
GRA                 SA5    O.BIT                                                 FCC0675     238     A
GRA                 LX1    -BA.BIP                                               FCC0675     239     A
GRA                 SB3    X1          BN = BI[BAW]                              FCC0675     240     A
GRA                 MX0    -BS.CLPCL                                             FCC0675     241     A
GRA                 SA2    X5+B3       BIW = BIT(BN)                             FCC0675     242     A
GRA                 SA3    M.PL                                                  FCC0675     243     A
GRA                 LX0    BS.CLPCP                                              FCC0675     244     A
GRA                 LX4    -BS.LENP                                              GRA        4949     A
GRA                 SX1    X4          LEN = LEN[BSW]                            GRA        4950     A
GRA                 LX4    BS.LENP                                               GRA        4951     A
GRA                 SA5    O.PIT                                                 GRA        4952     A
GRA                 BX4    -X0*X4                                                GRA        4953     A
GRA                 SX7    B1                                                    GRA        4954     A
GRA                 BX2    X0*X2                                                 GRA        4955     A
GRA                 LX7    BI.CBP                                                GRA        4956     A
GRA                 LX4    BI.PCP-BS.PCP                                         GRA        4957     A
GRA                 BX2    X4+X2       CLPC[BIW] = CLPC[BSW]                     GRA        4958     A
GRA                 IX0    X3-X1                                                 GRA        4959     A
GRA                 SA4    PL                                                    GRA        4960     A
GRA                 BX7    X7+X2       CB[BIW] = 1   */ INDICATE BLOCK CODED     GRA        4961     A
GRA                 SA7    A2                                                    GRA        4962     A
GRA                 PL     X0,WSC2     M.PL = MAX( M.PL , LEN )                  GRA        4963     A
GRA                 SX6    X1                                                    GRA        4964     A
GRA                 SA6    A3                                                    GRA        4965     A
GRA                                                                              GRA        4966     A
GRA        WSC2     IX6    X4+X5       FWAB = O.PIT + PL                         GRA        4967     A
GRA                 IX7    X4+X1       PL = PL + LEN                             GRA        4968     A
GRA                 SA7    A4+                                                   GRA        4969     A
GRA                 CALL   WFB         WFB( FWAB , LEN )  */ WRITE TO *BLK*      GRA        4970          I
          -CCGNC
GRA                 CALL   WFB#        WFB( FWAB , LEN )  */ WRITE TO *BLK*      CCGNC       667     A
GRA                 MX7    0                                                     GRA        4971     A
GRA                 EQ     WSC                                                   GRA        4972     A
GRA        IRA      TITLE  IRA - INSERT REGISTER ASSIGNMENTS IN BLOCKS           GRA        4973     A
GRA                 QUAL   IRA                                                   GRA        4974     A
GRA                                                                              GRA        4975     A
GRA       **        IRA - INSERT REGISTER ASSIGNMENTS IN TO BLOCKS               GRA        4976     A
GRA       *         MODIFY SEQUENCE TO REFLECT REGISTER ASSIGNMENTS MADE IN GRA, GRA        4977     A
GRA       *         CHANGE INCREMENTS ( IA,IS ) TO SHORT INSTRUCTIONS, ETC.      GRA        4978     A
GRA       *                                                                      GRA        4979     A
GRA       *         ENTRY  (NXS) = N. EXTRA SPACE NEEDED                         GRA        4980     A
GRA       *                RCT, NAA,ACT, TRD, LUL, LCV, LIH SET.                 GRA        4981     A
GRA       *                LINK WORD OF INSTRUCTION POINTS TO *RCT* ENTRY.       GRA        4982     A
GRA       *                                                                      GRA        4983     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA     *         EXIT   (LBJ) " 0 , IF SEQUENCE CONTAINS A LOOP BACK JUMP          GRA        4984    A
GRA     *                                                                           GRA        4985    A
GRA     *         DURING *IRA* THE LINK WORDS HOLD THE R-NUMBERS OF THE             GRA        4986    A
GRA     *         INSTRUCTIONS THAT REFLECT THE VALUES THAT ARE AVAILABLE IN        GRA        4987    A
GRA     *         THE DIFFERENT REGISTER CLASS"S.                                   GRA        4988    A
GRA                                                                                 GRA        4989    A
GRA               DESCRIBE I.,60     RNI(INB,ST,BR,XR)                              GRA        4990    A
GRA      INB      DEFINE 1           =1 IF BR R-NUMBER IS AVAIL                     GRA        4991    A
GRA               DEFINE 5                                                          GRA        4992          I
         -CCG0034
GRA      MIB      DEFINE 1           MINUS VALUE IN B-REG                           CCG0034       6    A
GRA               DEFINE 4                                                          CCG0034       7    A
GRA      ST       DEFINE 18          ST REG R-NUMBER                                GRA        4993    A
GRA      BR       DEFINE 18          B-REG R-NUMBER                                 GRA        4994    A
GRA      XR       DEFINE 18          X-REG R-NUMBER                                 GRA        4995    A
GRA                                                                                 GRA        4996    A
GRA     *         REGISTER ASSIGNMENTS ( MAIN LOOP ) -                              GRA        4997    A
GRA     *         (X0) = R-NUMBER MASK, (A5) = R1 , (A1,X1) = RCT(J)                GRA        4998    A
GRA     *         (B2) = OC[R1] , B4 = [O.RCT] , B6 = TI , (B7) = [O.TXT]           GRA        4999    A
GRA      SB       MICRO  1,,/B5/     O.SEQ + 3                                      GRA        5000    A
GRA      IRA      SPACE  2                                                          GRA        5001    A
GRA      IRA      ROUTINE                                                           GRA        5002    A
GRA               SA1    L.SEQ                                                      GRA        5003    A
GRA               SA2    NXS                                                        GRA        5004    A
GRA               IX1    X1+X2                                                      GRA        5005    A
GRA               ALLOC  TXT,X1     ALLOC( TXT , L.SEQ+NXS )                        GRA        5006    A
GRA               SA4    O.SEQ      TI = OLD(L.TXT)                                 GRA        5007    A
GRA               SA3    O.RCT                                                      GRA        5008    A
GRA               SB7    X2         TB = O.TXT                                      GRA        5009    A
GRA               MX7    0                                                          GRA        5010    A
GRA               S"SB"  X4+3       SB = O.SEQ + 3                                  GRA        5011    A
GRA               SA5    X4         R1 = O.SEQ                                      GRA        5012    A
GRA               SA1    X4+4                                                       GRA        5013    A
GRA               SB4    X3         (B4) = [O.RCT]                                  GRA        5014    A
GRA               BX6    X5                                                         GRA        5015    A
GRA               SA6    B7         [TB] = [R1]   */ SET LINE NUM OF SEQ            GRA        5016    A
GRA               SA7    RVT        RVT(0) = 0                                      GRA        5017    A
GRA               SA7    LBJF       LBJF = 0                                        CCG0119      14    A
GRA               SA7    LBJ        LBJ = 0                                         GRA        5018          I
         -FCC0985
GRA               SA6    LBN        LBN = [R1]                                      FCC0985      21    A
GRA               SA7    "SB"       [SB] = 0                                        GRA        5019    A
GRA               UX6    B2,X1                                                      GRA        5020    A
GRA               SX7    B2-OC.LAB                                                  GRA        5021    A
GRA               SA0    4          LF = 4        */ INIT LAB NOT PRESENT          FCC2051      15    A
GRA               NZ     X7,IRA1    IF OC[R1+4] " OC.LAB                            GRA        5022    A
GRA                                                                                 GRA        5023    A
GRA               SA0    B0         LF = 0        */ INIT LAB PRESENT              FCC2051      16    A
GRA               SA5    A1         R1 = R1 + 4                                     GRA        5024    A
GRA               RJ     SRI        SRI(  LAB  [R1],0 )                             GRA        5025    A
GRA               SA1    F.RDT+OC.EOS                                               GRA        5026    A
GRA               BX6    X1         FT[DI] = 0                                      GRA        5027    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SA6    A6                                                    GRA        5028     A
GRA       IRA      SPACE  2,14                                                  GRA        5029     A
GRA      *         SCAN *RAT* AND SETUP INITIAL *DEF*"S AND *DAR*"S             GRA        5030     A
GRA      *         CA OF R2 WORD HOLDS THE *CA* IN THE B-REG FOR PLD/PST"S.     GRA        5031     A
GRA                                                                             GRA        5032     A
GRA       IRA1     SX5    27B         I = 27B       */ REGNO                    GRA        5033     A
GRA                SB3    OC.DEF                                                GRA        5034     A
GRA                MX0    IH.CAL                                                GRA        5035     A
GRA                SA1    =XF.RDT+OC.DEF                                        GRA        5036     A
GRA                LX0    IH.CAL+IH.CAP                                         GRA        5037     A
GRA                                                                             GRA        5038     A
GRA       IRA3     SA4    RAT+X5                                                GRA        5039     A
GRA                ZR     X4,IRA5     IF RAT(I) = 0                             GRA        5040     A
GRA                SA3    B4+X4       J = R1[RAT(I)];  CAW = RCT(J)             GRA        5041     A
GRA                SX6    B6                                                    GRA        5042     A
GRA                SA6    RVT+X5      RVT(I) = TI   */ R-NUMBER IN REG          GRA        5043     A
GRA                SX2    X5+SO.LOCK                                            GRA        5044     A
GRA                LX2    R1.SOP                                                GRA        5045     A
GRA                BX6    X2+X6                                                 GRA        5046     A
GRA                LX4    -RA.R3P                                               GRA        5047     A
GRA                PX6    B3,X6                                                 GRA        5048     A
GRA                SA6    B7+B6       TXT(TI) = TYII(OC.DEF,0,FL+I,TI)          GRA        5049     A
GRA                                                                             GRA        5050     A
GRA                SB2    X4+B1       J3 = R3[RAT(I)]                           GRA        5051     A
GRA                SA2    B4+B2                                                 GRA        5052     A
GRA                BX7    -X2*X0      CA[R2W] = -CA[RCT(J3+1)]                  GRA        5053     A
GRA                SA7    A6+B1       TXT(TI+1) = R2W                           GRA        5054     A
GRA                BX6    X1                                                    GRA        5055          I
          -FCC0603
GRA                SA6    A7+B1       TXT(TI+2) = RDT(OC.DEF)                   GRA        5056          I
          -FCC0603
GRA                MX7    0                                                     GRA        5057          I
          -FCC0603
GRA                SA7    A6+B1       TXT(TI+3) = 0                             GRA        5058          I
          -FCC0603
GRA                BX7    X1                                                    FCC0603      81     A
GRA                SA7    A7+B1       TXT(TI+2) = RDT(OC.DEF)                   FCC0603      82     A
GRA                MX7    0                                                     FCC0603      83     A
GRA                SA7    A7+B1       TXT(TI+3) = 0                             FCC0603      84     A
GRA                LX3    59-CA.VCP                                             GRA        5059     A
GRA                SB6    B6+4        TI = TI + 4                               GRA        5060     A
GRA                                                                             GRA        5061     A
GRA                PL     X3,IRA5     IF ^VC[CAW]                               GRA        5062     A
GRA                LX3    CA.VCP-CA.ACP                                         GRA        5063     A
GRA                PL     X3,IRA5     IF ^AC[CAW]   */ NOT A PREFETCH           GRA        5064     A
GRA                                                                             GRA        5065     A
GRA      *         OUTPUT *DAR* TO DEFINE THE R-NUM IN THE A-REG                GRA        5066     A
GRA                                                                             GRA        5067     A
GRA                SX7    B1                                                    FCC0603      85     A
GRA                LX7    R1.SOP+SO.LKP                                         FCC0603      86     A
GRA                BX6    X7+X6       SOLK[TXT(TI-4)] = 3  */ INDICATE *A-LOCK* FCC0603      87     A
GRA                SA6    A6                                                    FCC0603      88     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SX6    B6                                                    GRA        5068     A
GRA                SA6    RVT+X5-8    RVT(I-8) = TI                             CCG25       163     A
GRA                SX4    B6-4                                                  GRA        5069     A
GRA                TYPEI  DAR,X6,X4   SRI(  DAR  TI,TI-4 )                      GRA        5070     A
GRA                                                                             GRA        5071     A
GRA       IRA5     SX5    X5-1        I = I - 1                                 GRA        5072     A
GRA                NZ     X5,IRA3     IF I " 0                                  GRA        5073     A
GRA                                                                             GRA        5074     A
GRA                MX0    -R1.RIL+1                                             GRA        5075          I
          -CCG11
GRA                LX0    1                                                     GRA        5076          I
          -CCG11
GRA                SX6    B6+A0       DTL = TI + LF                             FCC2051      17     A
GRA                SA6    DTL                                                   FCC2051      18     A
GRA                SX0    RN.MASK                                               CCG11       121     A
GRA                EQ     IRA10                                                 GRA        5077     A
GRA       EOQ      SPACE  3,20                                                  GRA        5078     A
GRA                PROCESS EOQ                                                  GRA        5079     A
GRA                SA1    LBJ                                                   GRA        5080          I
          -FCC0985
GRA                SA2    EPSI                                                  GRA        5081          I
          -FCC0985
GRA                SA3    NPS                                                   GRA        5082          I
          -FCC0985
GRA                ZR     X1,EOQ5     IF LBJ = 0    */ NO LOOP BACK JUMP        GRA        5083          I
          -FCC0985
GRA                ZR     X2,EOQ5     IF EPSI = 0   */ NO EPILOGUE POST STORES  GRA        5084          I
          -FCC0985
GRA                ZR     X3,EOQ5     IF NPS = 0    */ NO POST STORES           GRA        5085          I
          -FCC0985
GRA                                                                             FCC0513      10          I
          -FCC0985
GRA       *        OUTPUT EPILOGUE POST STORES FOR VARIABLES LIVE ON EXIT       FCC0513      11          I
          -FCC0985
GRA       *        FROM THE LATCHING NODE.                                      FCC0513      12          I
          -FCC0985
GRA                                                                             GRA        5086          I
          -FCC0985
GRA                SA1    O.RXI                                                 GRA        5087          I
          -FCC0985
GRA                SA4    O.UDT                                                 GRA        5088          I
          -FCC0985
GRA                SA0    X3          I = NPS                                   GRA        5089          I
          -FCC0985
GRA                SB3    X4+B1       (B3) = [O.UDT] + 1                        GRA        5090          I
          -FCC0985
GRA                IX6    X1+X2                                                 GRA        5091          I
          -FCC0985
GRA                SB5    X6          RB = O.RXI + EPSI  */ LIVE ENTRY BIT VECTOR GRA      5092          I
          -FCC0985
GRA                SA3    NPS                                                   FCC0985      22     A
GRA                SA2    EPSI                                                  FCC0985      23     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                ZR     X3,EOQ5     IF NPS = 0                                  FCC0985    24    A
GRA                ZR     X2,EOQ5     IF EPSI = 0                                 FCC0985    25    A
GRA                SA1    A5-4        R1 = R1 - 4                                 FCC0985    26          I
          -FCC2051
GRA                SB3    OC.JPX                                                  FCC0985    27          I
          -FCC2051
GRA                UX6    B2,X1                                                   FCC0985    28          I
          -FCC2051
GRA                NE     B2,B3,EOQ5  IF OC[R1] " OC.JPX  */ LAST NOT A *JPX*    FCC0985    29          I
          -FCC2051
GRA                                                                               FCC0985    30    A
GRA       *        SEARCH *RXI* FOR THIS *BN* AND SEE IF AN *EPS* NODE            FCC0985    31    A
GRA                                                                               FCC0985    32    A
GRA                SA5    BVL                                                     FCC0985    33    A
GRA                SA4    O.UDT                                                   FCC0985    34    A
GRA                SB2    X5+2        V = BVL + 2   */ RXI INCREMENT              FCC0985    35    A
GRA                SA1    O.RXI                                                   FCC0985    36    A
GRA                SB3    X4+B1       (B3) = UDT + 1                              FCC0985    37    A
GRA                SA0    X3          I = NPS                                     FCC0985    38    A
GRA                SA5    X1+B1       TI = RXI + 1                                FCC0985    39    A
GRA                MX1    -E.PREDL                                                FCC0985    40    A
GRA                SA2    LBN         BN = IH[LBN]  */ LAST BOS/EOS               FCC0985    41    A
GRA                                                                               FCC0985    42    A
GRA        EOQ1A   LX5    -E.PREDP                                                FCC0985    43    A
GRA                BX6    X2-X5                                                   FCC0985    44    A
GRA                LX5    59-E.SCRP+E.PREDP                                       FCC0985    45    A
GRA                BX7    -X1*X6                                                  FCC0985    46    A
GRA                ZR     X7,EOQ1B    IF PRED[RI] = BN                            FCC0985    47          I
          -FCC1107
GRA                SA5    A5+B2       RI = RI + V                                 FCC0985    48          I
          -FCC1107
GRA                NZ     X7,EOQ1B    IF PRED[RI] " BN                            FCC1107     5    A
GRA                PL     X5,EOQ1B    IF ^SCR[RI]                                 FCC1107     6    A
GRA                LX5    E.SCRP-E.JPP                                            FCC1107     7    A
GRA                PL     X5,EOQ1C    IF ^JP[RI]    */ A FALL THROUGH EDGE        FCC1107     8    A
GRA                                                                               FCC1107     9    A
GRA        EOQ1B   SA5    A5+B2       RI = RI + V                                 FCC1107    10    A
GRA                NZ     X5,EOQ1A    IF [RI] " 0   */ NOT END OF TABLE           FCC0985    49    A
GRA                EQ     EOQ5                                                    FCC0985    50    A
GRA                                                                               FCC0985    51    A
GRA        EOQ1B   PL     X5,EOQ5     IF ^SCR[RI]                                 FCC0985    52          I
          -FCC1107
GRA                LX5    E.SCRP-E.JPP                                            FCC0985    53          I
          -FCC1107
GRA                MI     X5,EOQ5     IF JP[RI]     */ ^ A FALL THROUGH EDGE      FCC0985    54          I
          -FCC1107
GRA                SB5    A5+1        RB = RI + 1   */ LIVE ENTRY BIT VECTOR      FCC0985    55          I
          -FCC1107
GRA        EOQ1C   SB5    A5+1        RB = RI + 1   */ LIVE ENTRY BIT VECTOR      FCC1107    11    A
GRA                                                                               FCC0985    56    A
GRA       *        OUTPUT EPILOGUE POST STORES FOR VARIABLES LIVE ON EXIT         FCC0985    57    A
GRA       *        FROM THIS BLOCK.                                               FCC0985    58    A

          0        1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                                                                                GRA        5093    A
GRA      EOQ1    SA1    PST-1+A0    J = CO[PST(I)]                                GRA        5094    A
GRA              SA2    B4+X1       CAW = RCT(J);  K = UDI[RCT(J)]                GRA        5095    A
GRA              LX1    -18         REGN = REG[PST(I)]                            GRA        5096    A
GRA              SA3    B3+X2       U2 = UDT(K+1)                                 GRA        5097    A
GRA              SA4    B5+X3       LEW = [RB+WI[U2]]                             GRA        5098    A
GRA              UX7    B2,X3       BIT = BITN[U2]                                GRA        5099        I
         -FCC0513
GRA              SB2    B2-59                                                     GRA        5100        I
         -FCC0513
GRA              LX7    -B2,X4                                                    GRA        5101        I
         -FCC0513
GRA              PL     X7,EOQ4     IF SHIFT(59-BIT,LEW) > 0  */ DEAD ON ENTRY    GRA        5102        I
         -FCC0513
GRA                                                                               GRA        5103        I
         -FCC0513
GRA      *       OUTPUT *ST* TO UPDATE VALUE IN MEMORY                            GRA        5104        I
         -FCC0513
GRA              SX6    B1                                                        FCC0513      13    A
GRA              UX3    B2,X3       B = BITN[U2]                                  FCC0513      14    A
GRA              LX6    B2,X6                                                     FCC0513      15    A
GRA              BX7    X6*X4                                                     FCC0513      16    A
GRA              ZR     X7,EOQ4     IF ^B[LEW]    */ DEAD ON ENTRY                FCC0513      17    A
GRA                                                                               FCC0513      18    A
GRA      *       SET BIT IN *LUV*, OUTPUT *ST* TO UPDATE VALUE IN MEMORY.         FCC0513      19    A
GRA                                                                               FCC0513      20    A
GRA              SA4    LUV                                                       FCC0513      21    A
GRA              SB2    X3                                                        FCC0513      22    A
GRA              SA4    X4+B2                                                     FCC0513      23    A
GRA              BX6    X6+X4       B[LUV(WI[U2])] = 1  */ SET BIT IN *LUV*       FCC0513      24    A
GRA              SA6    A4                                                        FCC0513      25    A
GRA                                                                               GRA        5105    A
GRA              SA4    RVT+X1      RN = RVT(REGN)                                GRA        5106    A
GRA              SX7    X1-26B                                                    GRA        5107    A
GRA              PL     X7,EOQ3     IF REGN \ 26B */ RESULT IN A *ST-REG*         GRA        5108    A
GRA                                                                               GRA        5109    A
GRA              SX7    X1-20B                                                    GRA        5110    A
GRA              SB2    OC.XMT      OC = OC.XMT                                   GRA        5111    A
GRA              PL     X7,EOQ2     IF REGN \ 20B */ IN A X-REG                   GRA        5112    A
GRA              SB2    OC.SA       OC = OC.SA                                    GRA        5113    A
GRA      EOQ2    MX7    0                                                         GRA        5114    A
GRA              TYPEI  B2,B6,X4    SRI(  XMT/SA  TI,RN )                         GRA        5115    A
GRA              SX4    B6-4        RN = TI - 4                                   GRA        5116    A
GRA                                                                               GRA        5117    A
GRA      EOQ3    SA2    A2+B1                                                     GRA        5118    A
GRA              SB2    OC.ST                                                     GRA        5119    A
GRA              MX1    -IH.CAIHL                                                 GRA        5120    A
GRA              SX6    X4          R1W = TYII(OC,0,0,RN)                         GRA        5121    A
GRA              BX7    -X1*X2      R2W = IHW(0,0,CAIH[RCT(J+1)])                 GRA        5122    A
GRA              RJ     SRI         SRI(  ST  RN,CAIH )                           GRA        5123    A
GRA                                                                               GRA        5124    A
GRA      EOQ4    SB2    A0-B1                                                     GRA        5125    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SA0    A0-B1       I = I - 1                                  GRA        5126    A
GRA                GT     B2,EOQ1     IF I > 0                                   GRA        5127    A
GRA                                                                              GRA        5128    A
GRA       EOQ5     SB2    OC.EOQ                                                 GRA        5129          I
         -CCG25
GRA      *         CHECK FOR AND OUTPUT STORES FOR THE A/X CANIDATES              CCG25       164    A
GRA                                                                              CCG25       165    A
GRA       EOQ5     SA1    AXCT        I = 1                                      CCG25       166          I
         -CCG0119
GRA       EOQ5     SA1    LBJF                                                   CCG0119      15    A
GRA                ZR     X1,EOQ7     IF NOT EXIT BLOCK                          CCG0119      16    A
GRA                                                                              CCG0119      17    A
GRA                SA1    AXCT        I = 1                                      CCG0119      18    A
GRA                                                                              CCG25       167    A
GRA       EOQ6     ZR     X1,EOQ7     IF AXCT(I) = 0                             CCG25       168    A
GRA                SA2    RVT+X1                                                 CCG25       169    A
GRA                MX7    0                                                      CCG25       170    A
GRA                TYPEI  XMT,B6,X2   SRI( XMT  RI,RVT(REGN) )                   CCG25       171    A
GRA                SA3    RVT+X1-10B                */ R-NUMBER IN A-REG          CCG25       172    A
GRA                AX1    18                                                     CCG25       173    A
GRA                SX1    X1+B1                                                  CCG25       174    A
GRA                SA2    B4+X1       C2W = RCT(CO+1)                            CCG25       175    A
GRA                MX7    -IH.CAIHL                                              CCG25       176    A
GRA                BX7    -X7*X2                                                 CCG25       177    A
GRA                SX6    B6-4                                                   CCG25       178    A
GRA                TYPEI  SST,X6,X3   SRI( SST RI-4,ARN,,CAIH )                  CCG25       179    A
GRA                SA1    A1+B1       I = I + 1                                  CCG25       180    A
GRA                EQ     EOQ6                                                   CCG25       181    A
GRA                                                                              CCG25       182    A
GRA       EOQ7     SB2    OC.EOQ                                                 CCG25       183    A
GRA                MX6    0                                                      GRA        5130    A
GRA                SX7    B0                                                     GRA        5131    A
GRA                CALL   SRI         SRI(  EOQ  0,0 )                           GRA        5132    A
GRA                SX6    B6+                                                    GRA        5133    A
GRA                SA6    L.TXT       L.TXT = TI                                 GRA        5134    A
GRA                EQ     IRA                                                    GRA        5135    A
GRA       TYI      SPACE  3,14                                                   GRA        5136          I
         -FCC0513
GRA       TYI      EJECT                                                         FCC0513      26    A
GRA                PROCESS (UP,NR,RNZ)                                           GRA        5137    A
GRA                ZR     X7,TYI      IF RJ[R1] = 0                              GRA        5138    A
GRA                SX2    B6+4        LJW = TI + 4  */ FAKE A B-OPERAND          GRA        5139    A
GRA                                                                              GRA        5140    A
GRA      *         TYPE I INSTRUCTION, 2 X OPERANDS AND ARBITRARY X-RESULT        GRA        5141    A
GRA                                                                              GRA        5142    A
GRA       TYI      SX2    X2          J = SETX(LJW)  */ X-REG OP                 GRA        5143    A
GRA                SX5    B6                                                     GRA        5144    A
GRA                LX2    R1.RJP                                                 GRA        5145    A
GRA                SX3    X3          K = SETX(LKW)                              GRA        5146    A
GRA                BX2    X2+X5                                                  GRA        5147    A
GRA                SX6    B6                                                     GRA        5148    A
GRA                LX5    I.STP                                                  GRA        5149    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                BX6    X5+X6                                                GRA        5150    A
GRA                SA6    A5+3        [R1+3] = RNI(0,TI,0,TI)                  GRA        5151    A
GRA                LX3    R1.RKP                                               GRA        5152    A
GRA                BX5    X2+X3                                                GRA        5153    A
GRA                                                                            GRA        5154    A
GRA        TYIA    MX7    0           R2W = 0                                  GRA        5155    A
GRA                SA4    F.RDT+B2    D = RDT(OC[R1])                          GRA        5156    A
GRA                PX6    B2,X5       R1W = TYI(OC[R1],J,K,TI)                 GRA        5157    A
GRA                                                                            GRA        5158    A
GRA        *       TRANSFER INSTRUCTION TO *TXT*,  (X4,X6,X7) = D,R1W,R2W     GRA        5159    A
GRA                                                                            GRA        5160    A
GRA        TYIT    SA6    B7+B6       TXT(TI) = R1W                            GRA        5161    A
GRA                SA7    A6+B1       TXT(TI+1) = R2W                          GRA        5162    A
GRA                BX6    X4                                                   GRA        5163    A
GRA                SA6    A7+B1       TXT(TI+2) = D                            GRA        5164    A
GRA                MX7    0                                                    GRA        5165    A
GRA                SA7    A6+B1       TXT(TI+3) = 0                            GRA        5166    A
GRA                SB6    B6+4        TI = TI + 4                              GRA        5167    A
GRA                                                                            GRA        5168    A
GRA        *       ADVANCE TO NEXT INSTRUCTION, JUMP TO OPCODE PROCESSOR      GRA        5169    A
GRA                                                                            GRA        5170    A
GRA                PROCESS (BOS,NOP)                                           GRA        5171    A
GRA        IRA10   SA5    A5+4        R1 = R1 + 4                              GRA        5172    A
GRA                SA4    A5+2        DI = R1 + 2                              GRA        5173    A
GRA                MX7    D.TYL                                                GRA        5174    A
GRA                SA1    A4+B1       LI = DI + 1                              GRA        5175    A
GRA                LX4    58-D.TYP                                             GRA        5176    A
GRA                BX3    X7*X4                                                GRA        5177    A
GRA                UX6    B2,X5                                                GRA        5178    A
GRA                NZ     X3,IRA11    IF TYPE[DI] " I                          GRA        5179    A
GRA                                                                            GRA        5180    A
GRA                LX6    -R1.RJP                                              GRA        5181    A
GRA                BX7    "RN"X6                                               GRA        5182    A
GRA                LX6    R1.RJP-R1.RKP                                        GRA        5183    A
GRA                SA2    "SB"+X7     LJW = [SB+RJ[R1]]                        GRA        5184    A
GRA                BX6    "RN"X6                                               GRA        5185    A
GRA                SA3    "SB"+X6     LKW = [SB+RK[R1]]                        GRA        5186    A
GRA                                                                            GRA        5187    A
GRA        IRA11   SA1    B4+X1       J = CO[LI];  CAW = RCT(J)                GRA        5188    A
GRA                LX4    2+D.TYP                                              GRA        5189    A
GRA                JP     IRA.JT+B2   JUMP( IRA.JT( OC[R1] ) )                 GRA        5190    A
GRA        TYIV    SPACE  3,14                                                 GRA        5191    A
GRA        *       TYPE IV INSTRUCTIONS                                       GRA        5192    A
GRA                                                                            GRA        5193    A
GRA                PROCESS EOS                                                 GRA        5194    A
GRA                SX3    X5                                                   GRA        5195         I
           -FCC0985
GRA                NZ     X3,IRA10    IF IH[R1] " 0  */ BASIC BLOCK MARKER     GRA        5196         I
           -FCC0985
GRA                SX6    X5                                                   FCC0985      59    A
GRA                ZR     X6,.LAB     IF IH[R1] = 0 */ ^ OPT=2 *EOS*           FCC0985      60    A
GRA                                                                            FCC0985      61    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SA6    LBN         LBN = IH[R1]  */ SAVE FOR *EOQ*          FCC0985      62    A
GRA                EQ     IRA10                                                 FCC0985      63    A
GRA                                                                             GRA         5197    A
GRA                PROCESS (LAB,ENT,UJP,RJ3,RJ6)                                GRA         5198    A
GRA                SA3    A5+B1                                                 GRA         5199    A
GRA                LX6    X5                                                    GRA         5200    A
GRA                BX7    X3          R2W = [R1+1]                              GRA         5201    A
GRA                EQ     TYIT                                                  GRA         5202    A
GRA         TYIM   SPACE  3,14                                                  GRA         5203    A
GRA        *       MISCELLANEOUS TYPE I INSTRUCTIONS                            GRA         5204    A
GRA                                                                             GRA         5205    A
GRA                PROCESS (ILS,IRS,PK)                                         GRA         5206    A
GRA                LX2    -I.BRP      LJW = SHIFT(LJW,-I.BRP)                   GRA         5207    A
GRA                EQ     TYI                                                   GRA         5208    A
GRA         DRL    SPACE  2,14                                                  GRA         5209    A
GRA                PROCESS DRL                                                  GRA         5210    A
GRA                SX7    B6                                                    GRA         5211          I
           -CCG0045
GRA                SX6    B6          R = TI                                    GRA         5212          I
           -CCG0045
GRA                SX6    B6          R = TI                                    CCG0045        6    A
GRA                BX7    X4                                                    CCG0045        7    A
GRA                SA6    A4+B1       [DI+1] = RNI(0,0,0,TI)                    CCG0045        8    A
GRA                LX7    59-D.L2P                                              CCG0045        9    A
GRA                PL     X7,DWL1     IF ^L2[DI]    */ LEVEL 0                  CCG0045       10    A
GRA                SX7    B6          R = TI                                    CCG0045       11    A
GRA                LX7    I.STP                                                 GRA         5213    A
GRA                BX7    X6+X7                                                 GRA         5214    A
GRA                SA7    A4+B1       [DI+1] = RNI(0,TI,0,TI)                   GRA         5215    A
GRA                EQ     DWL1                                                  GRA         5216    A
GRA                                                                             GRA         5217    A
GRA                PROCESS DWL                                                  GRA         5218    A
GRA                BX7    "RN"X5                                                GRA         5219    A
GRA                SA3    "SB"+X7                                               GRA         5220    A
GRA                SX6    X3+         R = XR[SB+RI[R1]]                         GRA         5221          I
           -CCGA039
GRA                BX7    X4
>>>>       CCGA039 11                                                           CCGA039        5    A
GRA                LX7    59-D.L2P
>>>>       CCGA039 12                                                           CCGA039        6    A
GRA                MI     X7,DWL0     IF L2[DI]
>>>>       CCGA039 13                                                           CCGA039        7    A
GRA                LX3    -I.STP      R = ST[SB+RI[R1]]
>>>>       CCGA039 14                                                           CCGA039        8    A
GRA        *
>>>>       CCGA039 15                                                           CCGA039        9    A
GRA         DWL0   SX6    X3          ELSE R= XR[SB+RI[R1]]
>>>>       CCGA039 16                                                           CCGA039       10    A
GRA                                                                             GRA         5222    A
GRA         DWL1   SX2    X2          J = XR[LJW]                               GRA         5223    A
GRA                SA1    A5+B1       R2 = R1 + 1                               GRA         5224    A
GRA                LX2    R1.RJP                                                GRA         5225    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                 BX6    X2+X6                                          GRA        5226    A
GRA                 LX1    -IH.RFP                                        GRA        5227    A
GRA                 PX6    B2,X6                                          GRA        5228    A
GRA                 SA2    "SB"+X1     LF = [SB+RF[R2]]                   GRA        5229         I
          -FCC0645
GRA                 BX1    X0*X1                                          GRA        5230         I
          -FCC0645
GRA                 PL     X2,DWL2     IF ^INB[LF]                        GRA        5231         I
          -FCC0645
GRA                 LX2    -I.BRP      LK = SHIFT(LF,-I.BRP)              GRA        5232         I
          -FCC0645
GRA        DWL2     SX2    X2                                             GRA        5233         I
          -FCC0645
GRA                 BX7    X2+X1       RF[R2] = SETX(LF)                  GRA        5234         I
          -FCC0645
GRA                 BX3    "RN"X1      LF = RF[R2]                        FCC0645      22    A
GRA                 BX1    X0*X1                                          FCC0645      23    A
GRA                 ZR     X3,DWL2     IF LF = 0     */ NO RF             FCC0645      24    A
GRA                 SA3    "SB"+X3     LF = [SB+R]                        FCC0645      25    A
GRA                 NZ     X3,DWL2     IF LF " 0     */ RF MATERIALIZED   FCC0645      26    A
GRA                 LX2    -R1.RJP                                        FCC0645      27    A
GRA                 SX3    X2          LF = J        */ USE RJ AS RF      FCC0645      28    A
GRA        DWL2     PL     X3,DWL3     IF ^INB[LF]                        FCC0645      29    A
GRA                 LX3    -I.BRP      LF = SHIFT(LF,-I.BRP)              FCC0645      30    A
GRA        DWL3     SX3    X3                                             FCC0645      31    A
GRA                 BX7    X3+X1       RF[R2] = SETX(LF)                  FCC0645      32    A
GRA                 SA4    F.RDT+B2    D = RDT(OC[R1])                    GRA        5235         I
          -CCG16
GRA                 SA3    F.RDT+B2                                       CCG16       235    A
GRA                 SX2    B1                                             CCG16       236    A
GRA                 LX2    D.L2P                     */ L2[D] = 0 IF *SUB0*  CCG16  237    A
GRA                 BX4    -X4*X2                                         CCG16       238    A
GRA                 BX4    X3-X4       D = RDT(OC[R1]) & ^L2[DI]          CCG16       239    A
GRA                 LX7    IH.RFP                                         GRA        5236    A
GRA                 EQ     TYIT                                           GRA        5237    A
GRA        SS       SPACE  2,6                                            GRA        5238    A
GRA                 PROCESS SS                                            GRA        5239    A
GRA                 LX2    -I.BRP                                         GRA        5240    A
GRA                 LX3    -I.BRP                                         GRA        5241    A
GRA                 EQ     TYI                                            GRA        5242    A
GRA        SA       SPACE  2,14                                           GRA        5243    A
GRA                 PROCESS SA                                            GRA        5244    A
GRA                 LX1    59-CA.RAP                                      GRA        5245    A
GRA                 MI     X1,PVC      IF RA[CAW]                         GRA        5246    A
GRA                 PL     X2,SA1      IF ^INB[LJW]                       GRA        5247    A
GRA                 LX2    -I.BRP      LJW = SHIFT(LJW,-I.BRP)            GRA        5248    A
GRA                 NZ     X3,SA1      IF LKW " 0                         GRA        5249    A
GRA                                                                       GRA        5250    A
GRA                 LX4    59-D.PRSP                                      GRA        5251    A
GRA                 MX7    -SO.REGL                                       GRA        5252    A
GRA                 PL     X4,SA1      IF ^PRS[DI]                        GRA        5253         I
          -FCC0414

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                PL     X4,SA2       IF ^PRS[DI]                                FCC0414      10     A
GRA                SA1    A5+4         R1RS = R1 + 4                              GRA        5254     A
GRA                LX1    -R1.SOP                                                 GRA        5255     A
GRA                BX6    -X7*X1       REGNO = SOREG[R1RS]                        GRA        5256     A
GRA                LX1    59-SO.LKP                                               GRA        5257     A
GRA                SX7    X6-8                                                    GRA        5258     A
GRA                PL     X7,SA1       IF REGNO > 7                               GRA        5259     A
GRA                PL     X1,SA1       IF SOLK[R1RS] " 1  */ NOT TEMP LOCK        GRA        5260     A
GRA                                                                               GRA        5261     A
GRA                SA5    A1           R = R1RS      */ OMIT *RS*                 GRA        5262     A
GRA                LX2    I.BRP                                                   GRA        5263     A
GRA                BX7    X2                                                      GRA        5264     A
GRA                SA7    A1-B1        [R1+3] = LJW                               GRA        5265     A
GRA                EQ     IRA10                                                   GRA        5266     A
GRA                                                                               GRA        5267     A
GRA        SA1     PL     X3,TYI       IF ^INB[LKW]                               GRA        5268     A
GRA                LX3    -I.BRP       LKW = SHIFT(LKW,-I.BRP)                    GRA        5269     A
GRA                EQ     TYI                                                     GRA        5270     A
GRA                                                                               FCC0414      11     A
GRA      *         SA  RI,RJ ,AND OPERAND IN A B-REG                             FCC0414      12     A
GRA                                                                               FCC0414      13     A
GRA        SA2     SX6    B6                                                      FCC0414      14     A
GRA                SX2    X2                                                      FCC0414      15     A
GRA                MX7    1                                                       FCC0414      16     A
GRA                LX2    I.BRP                                                   FCC0414      17     A
GRA                BX7    X7+X6                                                   FCC0414      18     A
GRA                LX6    I.STP                                                   FCC0414      19     A
GRA                BX6    X2+X6                                                   FCC0414      20     A
GRA                BX7    X7+X6                                                   FCC0414      21     A
GRA                SA7    A4+B1        [DI+1] = RNI(1,TI,SETX(LJW),TI)            FCC0414      22     A
GRA                SX5    B6                                                      FCC0414      23     A
GRA                LX2    R1.RJP-I.BRP                                            FCC0414      24     A
GRA                BX5    X2+X5                                                   FCC0414      25     A
GRA                EQ     TYIA                                                    FCC0414      26     A
GRA        IA      SPACE  2,14                                                    GRA        5271     A
GRA                PROCESS IA                                                     GRA        5272     A
GRA                BX6    X2+X3                                                   GRA        5273     A
GRA                PL     X6,TYI       IF ^( INB[LJW] ! INB[LKW] )                GRA        5274     A
GRA                LX1    59-CA.RFP                                               GRA        5275          I
          -CCGA097
GRA                SA4    =XHO$LCM                                                CCGA097       8     A
GRA                PL     X4,IAA            */ IF NOT LCM=G                       CCGA097       9     A
GRA                SA4    A2-3         R1W = [SB+RJ[R1]] - 3                      CCGA097      10     A
GRA                UX4    B2                                                      CCGA097      11     A
GRA                PL     X2,IA4       IF ^INB[LJW]                               CCGA097      12     A
GRA                SX7    -OC.DEF                                                 CCGA097      13     A
GRA                SX7    X7+B2                                                   CCGA097      14     A
GRA                NZ     X7,IA4       IF ^(OC = OC.DEF)                          CCGA097      15     A
GRA                SX3    B6                                                      CCGA097      16     A
GRA                EQ     IA6                                                     CCGA097      17     A
GRA                                                                               CCGA097      18     A
GRA        IA4     SA4    A3-3         R1W = [SB+RK[R1]] - 3                      CCGA097      19     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                UX4    B2                                                    CCGA097      20    A
GRA                MI     X3,IA5      IF INB[LKW]                               CCGA097      21    A
GRA                                                                              CCGA097      22    A
GRA        IA      SB2    OC.IA                            */   RESTORE B2      CCGA097      23    A
GRA                                                                              CCGA097      24    A
GRA        IAA     LX1    59-CA.RFP                                             CCGA097      25    A
GRA                PL     X1,TYI      IF ^RF[CI]    */ ^ *INC* AND SHORT USE    GRA        5276    A
GRA                                                                              GRA        5277    A
GRA                SB2    OC.SA       OC = OC.SA                                GRA        5278    A
GRA                BX6    X2*X3                                                 GRA        5279    A
GRA                MI     X6,IS1      IF INB[LJW] & INB[LKW]                    GRA        5280         I
          -CCG0034
GRA                MI     X6,IA2      IF INB[LJW] AND INB[LKW]                  CCG0034       8    A
GRA                BX6    X2                                                    GRA        5281    A
GRA                PL     X2,IA1      IF ^INB[LJW]  */ X+B IS OK                GRA        5282    A
GRA                BX2    X3                                                    GRA        5283    A
GRA                LX3    X6          SWAP( LJW , LKW )                         GRA        5284    A
GRA        IA1     LX3    -I.BRP      LKW = SHIFT(LKW,-I.BRP)                   GRA        5285    A
GRA                EQ     TYI                                                   GRA        5286    A
GRA        IA2     BX6    X2+X3                                                 CCG0034       9    A
GRA                LX6    59-I.MIBP                                             CCG0034      10    A
GRA                PL     X6,IS1      IF NOT ( MIB[LJW] OR MIB]LKW] )           CCG0034      11    A
GRA                SB2    OC.SS       OC = OC.SS                                CCG0034      12    A
GRA                BX6    X3                                                    CCG0034      13    A
GRA                LX6    59-I.MIBP                                             CCG0034      14    A
GRA                MI     X6,IS1      IF MIB[LKW]                               CCG0034      15    A
GRA                BX3    X2          SWAP( LJW , LKW )                         CCG0034      16    A
GRA                LX6    1+I.MIBP                                              CCG0034      17    A
GRA                BX2    X6                                                    CCG0034      18    A
GRA                EQ     IS1                                                   CCG0034      19    A
GRA                                                                              CCGA097      26    A
GRA        IA5     SX7    -OC.DEF                                               CCGA097      27    A
GRA                SX7    X7+B2                                                 CCGA097      28    A
GRA                NZ     X7,IA       IF ^(OC = OC.DEF)                         CCGA097      29    A
GRA                SX3    B6                                                    CCGA097      30    A
GRA                                                                              CCGA097      31    A
GRA        IA6     LX4    -R1.SOP                                               CCGA097      32    A
GRA                MX7    -SO.REGL                                              CCGA097      33    A
GRA                BX5    -X7*X4                                                CCGA097      34    A
GRA                SB2    -4                                                    CCGA097      35    A
GRA                                                                              CCGA097      36    A
GRA        IA7     SX7    -OC.DEF                                               CCGA097      37    A
GRA                                                                              CCGA097      38    A
GRA        IA8     SB2    B2+4                                                  CCGA097      39    A
GRA                GT     B2,B6,IA                    */ IF END OF TABLE        CCGA097      40    A
GRA                SA4    B7+B2                                                 CCGA097      41    A
GRA                UX4    B3                                                    CCGA097      42    A
GRA                SX6    X7+B3                                                 CCGA097      43    A
GRA                NZ     X6,IA8                          */ IF NOT OC.DEF      CCGA097      44    A
GRA                MX7    -SO.REGL                                              CCGA097      45    A
GRA                LX4    -R1.SOP                                               CCGA097      46    A
GRA                BX7    -X7*X4                          */ REG TYPE AND NUM   CCGA097      47    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                IX7    X7-X5                                                  CCGA097      48    A
GRA                NZ     X7,IA7              */ IF NOT B REG WE ARE LOOKING FOR CCGA097      49    A
GRA                SX4    B2                          */ B REG RI                CCGA097      50    A
GRA                SX6    B6                          */ NEXT AVAILABLE R NUM    CCGA097      51    A
GRA                TYPEI  SA,X6,X4    SRI(SA, RI, J)                             CCGA097      52    A
GRA                EQ     IA                                                    CCGA097      53    A
GRA        IS      SPACE  2,8                                                   GRA        5287    A
GRA                PROCESS IS                                                   GRA        5288    A
GRA                BX6    X2*X3                                                 GRA        5289    A
GRA                PL     X6,TYI      IF ^( INB[LJW] & INB[LKW] )               GRA        5290    A
GRA                LX1    59-CA.RFP                                             GRA        5291    A
GRA                PL     X1,TYI      IF ^RF[CI]    */ ^ *INC* AND SHORT USE    GRA        5292    A
GRA                                                                             GRA        5293    A
GRA                SB2    OC.SS       OC = OC.SS                                GRA        5294    A
GRA                BX6    X3                                                    CCG0034      20    A
GRA                LX6    59-I.MIBP                                             CCG0034      21    A
GRA                PL     X6,IS1      IF NOT MIB[LKW]                           CCG0034      22    A
GRA                SB2    OC.SA       OC = OC.SA                                CCG0034      23    A
GRA        IS1     LX2    -I.BRP      LJW = SHIFT(LJW,-I.BRP)                   GRA        5295    A
GRA                LX3    -I.BRP      LKW = SHIFT(LKW,-I.BRP)                   GRA        5296    A
GRA                EQ     TYI                                                   GRA        5297    A
GRA        SXT     SPACE  2,14                                                  GRA        5298    A
GRA                PROCESS SXT                                                  GRA        5299    A
GRA                LX1    59-CA.RAP                                             GRA        5300    A
GRA                PL     X1,TYI      IF ^RA[CAW]                               GRA        5301    A
GRA                                                                             GRA        5302    A
GRA                SA1    A1+2        CCW = RCT(J+2)                            GRA        5303    A
GRA                SA5    A5+4        R1 = R1 + 4                               GRA        5304    A
GRA                MX7    -CC.REG1L                                             GRA        5305    A
GRA                SX3    X2          LKW = XR[LJW]                             GRA        5306    A
GRA                LX1    -CC.REG1P                                             GRA        5307    A
GRA                BX7    -X7*X1                                                GRA        5308    A
GRA                SA2    RVT+X7      LJW = RVT(REG1[CCW])                      GRA        5309    A
GRA                UX6    B2,X5                                                 GRA        5310    A
GRA                SB2    B2-OC.KLS+OC.ILS  OC = OC[R1]-OC.KLS+OC.ILS           GRA        5311    A
GRA                EQ     TYI                                                   GRA        5312    A
GRA        XMTC    SPACE  2                                                     FCC0513      27    A
GRA                PROCESS XMTC                                                 FCC0513      28    A
GRA                PL     X2,TYI      IF ^INB[LJW]                              FCC0513      29    A
GRA                SB2    OC.SS       OC = OC.SS                                FCC0513      30    A
GRA                LX2    -I.BRP                                                FCC0513      31    A
GRA                SX3    X2          LKW = BR[LJW]                             FCC0513      32    A
GRA                SX2    0           LJW = 0                                   FCC0513      33    A
GRA                EQ     TYI                                                   FCC0513      34    A
GRA        XMT     SPACE  3,14                                                  GRA        5313    A
GRA                PROCESS XMT                                                  GRA        5314    A
GRA                LX1    59-CA.RAP                                             GRA        5315    A
GRA                MI     X1,PVC      IF RA[CAW]                                GRA        5316    A
GRA                LX4    59-D.PSP                                              GRA        5317    A
GRA                BX7    X2                                                    GRA        5318    A
GRA                PL     X4,XMT1     IF ^PS[DI]                                GRA        5319    A
GRA                LX7    -I.STP                                                GRA        5320    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SX6    X7                                                    GRA          5321     A
GRA                ZR     X6,XMT1     IF ST[LJW] = 0  */ OPERAND NOT IN *ST* REG GRA          5322     A
GRA                                                                              GRA          5323     A
GRA      *         OPERAND FREE TO GO TO A *ST* REG, SKIP THE *XMT*              GRA          5324     A
GRA                                                                              GRA          5325     A
GRA                LX7    I.STP                                                 GRA          5326     A
GRA                SA7    A4+B1       [DI+1] = LJW                              GRA          5327     A
GRA                EQ     IRA10                                                 GRA          5328     A
GRA                                                                              GRA          5329     A
GRA       XMT1     SX3    B6                                                    GRA          5330     A
GRA                SX2    X2          J = XR[LJW]                               GRA          5331          I
          -FCC1740X
GRA                LX3    I.STP                                                 GRA          5332          I
          -FCC1740X
GRA                SX6    B6                                                    GRA          5333          I
          -FCC1740X
GRA                BX7    X3+X2                                                 GRA          5334          I
          -FCC1740X
GRA                LX2    R1.RJP                                                GRA          5335          I
          -FCC1740X
GRA                SA7    A4+B1       [DI+1] = RNI(0,TI,0,J)                    GRA          5336          I
          -FCC1740X
GRA                BX5    X2+X6       R1W = TYI(OC.XMT,J,0,TI)                  GRA          5337          I
          -FCC1740X
GRA                SX6    X2          J = XR[LJW]                               FCC1740X        5     A
GRA                LX3    I.STP                                                 FCC1740X        6     A
GRA                SX5    B6                                                    FCC1740X        7     A
GRA                BX7    X3+X6       LW = RNI(0,TI,0,J)                        FCC1740X        8     A
GRA                NZ     X6,XMT2     IF J " 0                                  FCC1740X        9     A
GRA                LX2    -I.BRP                                                FCC1740X       10     A
GRA                SX6    X2          J = BR[LJW]                               FCC1740X       11     A
GRA                SB2    OC.SA                                                 FCC1740X       12     A
GRA                BX7    X3+X5       LW = RNI(0,TI,0,TI)                       FCC1740X       13     A
GRA       XMT2     LX6    R1.RJP                                                FCC1740X       14     A
GRA                SA7    A4+1        [DI+1] = LW                               FCC1740X       15          I
          -CCG0012
GRA                LX4    D.PSP-D.PRSP                                          CCG0012         8     A
GRA                PL     X4,XMT3     IF NOT PRS[DI]                            CCG0012         9     A
GRA                SX7    X7          LW = RNI(0,0,0,X)  */ CLEAR ST FIELD      CCG0012        10     A
GRA                                                                              CCG0012        11     A
GRA       XMT3     SA7    A4+B1       [DI+1] = LW                               CCG0012        12     A
GRA                BX5    X6+X5       R1W = TYI(0,J,0,TI)                       FCC1740X       16     A
GRA                EQ     TYIA                                                  GRA          5338     A
GRA       PST      SPACE  2,8                                                   GRA          5339     A
GRA                PROCESS (PST,SST,SDS)                                        GRA          5340     A
GRA                BX7    "RN"X5                                                GRA          5341     A
GRA                SA1    "SB"+X7                                               GRA          5342     A
GRA                LX1    -I.STP                                                GRA          5343     A
GRA                SX6    X1          R = ST[SB+RI[R1]]                         GRA          5344     A
GRA                EQ     PLD1                                                  GRA          5345     A
GRA       PLD      SPACE  2,14                                                  GRA          5346     A
GRA                PROCESS (PLD,SLD,SDL)                                        GRA          5347     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SX6    B6+          R = TI                                          GRA        5348     A
GRA                SA6    A5+3         [R1+3] = RNI(0,0,0,TI)                          GRA        5349     A
GRA                                                                                    GRA        5350     A
GRA        PLD1    PL     X2,PLD2      IF ^INB[LJW]                                    GRA        5351     A
GRA                LX2    -I.BRP       LJW = SHIFT(LJW,-I.BRP)                         GRA        5352     A
GRA        PLD2    PL     X3,PLD3      IF ^INB[LKW]                                    GRA        5353     A
GRA                LX3    -I.BRP       LKW = SHIFT(LKW,-I.BRP)                         GRA        5354     A
GRA                                                                                    GRA        5355     A
GRA        PLD3    SX2    X2                                                           GRA        5356     A
GRA                PX6    B2,X6                                                        GRA        5357     A
GRA                LX2    R1.RJP                                                       GRA        5358     A
GRA                SX3    X3                                                           GRA        5359     A
GRA                BX6    X2+X6                                                        GRA        5360     A
GRA                SA2    A5+B1        R2 = R1 + 1                                     GRA        5361     A
GRA                LX3    R1.RKP                                                       GRA        5362     A
GRA                BX6    X3+X6        R1W = TYI(OC[R1],SETX(LJW),SETX(LKW),R)         GRA        5363     A
GRA                SA1    DMASK                                                        GRA        5364     A
GRA                LX2    -IH.RFP                                                      GRA        5365     A
GRA                BX5    "RN"X2                                                       GRA        5366     A
GRA                SA3    "SB"+X5      LFW = [SB+RF[R2]]                               GRA        5367     A
GRA                PL     X3,PLD4      IF ^INB[LFW]                                    GRA        5368     A
GRA                LX3    -I.BRP                                                       GRA        5369     A
GRA        PLD4    BX2    X0*X2                                                        GRA        5370     A
GRA                SX3    X3           RF[R2W] = SETX(LFW)                             GRA        5371     A
GRA                BX7    X2+X3                                                        GRA        5372     A
GRA                BX4    -X1*X4       D = [DI] & ^DMASK                               GRA        5373     A
GRA                LX7    IH.RFP                                                       GRA        5374     A
GRA                EQ     TYIT                                                         GRA        5375     A
GRA        RS      SPACE  2,14                                                         GRA        5376     A
GRA                PROCESS RS                                                          GRA        5377     A
GRA                MX7    -SO.REGL                                                     GRA        5378     A
GRA                LX6    -R1.SOP                                                      GRA        5379     A
GRA                BX2    -X7*X6       REGNO = SOREG[R1]                               GRA        5380     A
GRA                LX6    R1.SOP+59-R1.INP                                             GRA        5381     A
GRA                PL     X6,RS1       IF IN[R1] " 0 */ ^ RJ RS                        GRA        5382          I
          -FCC0414
GRA                PL     X6,RS1       IF IN[R1] = 0 */ ^ A RJRS                       FCC0414      27     A
GRA                SX6    B6           R = TI                                          GRA        5383     A
GRA                EQ     RS2                                                          GRA        5384     A
GRA                                                                                    GRA        5385     A
GRA        RS1     LX1    59-CA.RAP                                                    GRA        5386          I
          -FCC0811
GRA                MI     X1,IRA10     IF RA[CAW]    */ PRED ASSIGNED TO A REG         GRA        5387          I
          -FCC0811
GRA        RS1     LX1    59-CA.IRAP                                                   FCC0811      44     A
GRA                MI     X1,IRA10     IF IRA[CAW]   */ PRED AND RS TO BE MOVED         FCC0811      45     A
GRA                SX6    B6-4         R = TI - 4                                      GRA        5388     A
GRA                SX3    X2-8                                                         GRA        5389     A
GRA                PL     X3,RS3       IF REGNO > 7                                    GRA        5390          I
          -FCC0764
GRA                MI     X3,RS2       IF REGNO < 8  */ IN A B-REGISTER                FCC0764       5     A
GRA                SX3    X2-26B                                                       FCC0764       6     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                PL     X3,RS3      IF REGNO \ 26B  */ IN A ST-REGISTER      FCC0764        7     A
GRA                                                                            FCC0764        8     A
GRA                SA2    A5-B1       LIP = R1 - 1  */ LINK WORD OF RS PRED    FCC0764        9     A
GRA                SX7    X2                                                   FCC0764       10     A
GRA                LX2    -I.STP                                               FCC0764       11     A
GRA                SX3    X2                                                   FCC0764       12     A
GRA                BX7    X7-X3                                                FCC0764       13     A
GRA                NZ     X7,RS3      IF ST[LIP] " XR[LIP]                     FCC0764       14     A
GRA                BX7    X0*X2                                                FCC0764       15     A
GRA                LX7    I.STP       ST[LIP] = 0                              FCC0764       16     A
GRA                SA7    A2                                                   FCC0764       17     A
GRA                EQ     RS3                                                  FCC0764       18     A
GRA                                                                            GRA         5391     A
GRA        RS2     MX7    1                                                    GRA         5392     A
GRA                LX7    1+I.INBP                                             GRA         5393     A
GRA                LX6    I.BRP                                                GRA         5394     A
GRA                BX7    X6+X7                                                GRA         5395     A
GRA                LX6    -I.BRP                                               GRA         5396     A
GRA                BX3    "RN"X5                                               GRA         5397     A
GRA                SA7    "SB"+X3     [SB+RI[R1]] = RNI(B,0,R,0)               GRA         5398     A
GRA                                                                            GRA         5399     A
GRA        RS3     BX5    X0*X5                                                GRA         5400     A
GRA                MX2    -D.USESL                                             GRA         5401     A
GRA                BX6    X5+X6       RI[R1W] = R                              GRA         5402     A
GRA                MX7    0           R2W = 0                                  GRA         5403     A
GRA                BX4    X2*X4       USES[DI] = 0;  D = [DI]                  GRA         5404     A
GRA                EQ     TYIT                                                 GRA         5405     A
GRA        TYII    SPACE  3,14                                                 GRA         5406     A
GRA                PROCESS (FMA,CLR,S)                                         GRA         5407     A
GRA                LX1    59-CA.RAP                                            GRA         5408     A
GRA                MI     X1,PVC      IF RA[CAW]                               GRA         5409     A
GRA                                                                            GRA         5410     A
GRA        FMA1    SX7    B6                                                   GRA         5411     A
GRA                SX6    B6                                                   GRA         5412     A
GRA                LX7    I.STP                                                GRA         5413     A
GRA                BX6    X7+X6       RNW = RNI(0,TI,0,TI)                     GRA         5414     A
GRA                                                                            GRA         5415     A
GRA        FMA2    SA6    A4+B1       [DI+1] = RNW                             GRA         5416     A
GRA                BX5    X0*X5                                                GRA         5417     A
GRA                SX7    B6                                                   GRA         5418     A
GRA                BX5    X7+X5       RI[R1W] = TI                             GRA         5419     A
GRA                EQ     TYIA                                                 GRA         5420     A
GRA        DEF     SPACE  2                                                    GRA         5421     A
GRA                PROCESS DEF                                                 GRA         5422     A
GRA                MX3    -SO.REGL                                             GRA         5423     A
GRA                LX5    -R1.SOP                                              GRA         5424     A
GRA                BX7    -X3*X5      REGN = SOREG[R1]                         GRA         5425     A
GRA                LX5    R1.SOP                                               GRA         5426     A
GRA                SX2    X7-26B                                               GRA         5427           I
           -FCC0414
GRA                SA3    RVT+X7                                               FCC0414       28     A
GRA                SX2    X7-26B                                               FCC0414       29     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                NZ     X3,DEF1     IF RVT(REGN) " 0  */ DEF IS A CANIDATE          FCC0414      30     A
GRA                PL     X2,FMA1     IF REGN \ 26B  */ IN A STORE REG                GRA        5428     A
GRA                AX7    4                                                           GRA        5429     A
GRA                SX6    B6          RNW = RNI(0,0,0,TI)                             GRA        5430     A
GRA                NZ     X7,FMA2     IF REGN \ 20B   */ X-REG                        GRA        5431     A
GRA                MX7    1                                                           GRA        5432     A
GRA                LX6    I.BRP                                                       GRA        5433     A
GRA                BX6    X7+X6       RNW = RNI(1,0,TI,0)                             GRA        5434     A
GRA                EQ     FMA2                                                        GRA        5435     A
GRA                                                                                   FCC0414      31     A
GRA       *        REGISTER ASSIGNMENT EXTENDED FROM INNER LOOP, SETUP LINK WORD     FCC0414      32     A
GRA       *        AND SKIP THIS *DEF*.                                              FCC0414      33     A
GRA                                                                                   FCC0414      34     A
GRA        DEF1    SX6    X3          R = RVT(REGN)                                   FCC0414      35     A
GRA                LX6    I.STP                                                       FCC0414      36     A
GRA                BX6    X3+X6       RNW = RNI(0,R,0,R)                              FCC0414      37     A
GRA                AX7    4                                                           FCC0414      38     A
GRA                PL     X2,DEF2     IF REGN \ 26B */ STORE REG                      FCC0414      39     A
GRA                MX2    1                                                           FCC0414      40     A
GRA                SX6    X3          RNW = RNI(0,0,0,R)                              FCC0414      41     A
GRA                NZ     X7,DEF2     IF REGN \ 20B */ X-REG                          FCC0414      42     A
GRA                LX3    I.BRP                                                       FCC0414      43     A
GRA                BX6    X2+X3       RNW = RNI(1,0,R,0)                              FCC0414      44     A
GRA        DEF2    SA6    A4+B1       [DI+1] = RNW                                    FCC0414      45     A
GRA                EQ     IRA10                                                       FCC0414      46     A
GRA        TYIII   SPACE  2,14                                                        GRA        5436     A
GRA                PROCESS (KLS,KRS)                                                  GRA        5437     A
GRA                SX7    B6                                                          GRA        5438     A
GRA                PX6    B2,X7       R1W = TYII(OC[R1],0,0,TI)                       GRA        5439     A
GRA                SA7    A4+B1       [DI+1] = RNI(0,0,0,TI)                          GRA        5440     A
GRA                SB3    B0          SC = 0                                          GRA        5441     A
GRA                                                                                   GRA        5442     A
GRA       *        TYIII - ADJUST *RF*, SC, R1W SET                                  GRA        5443     A
GRA                                                                                   GRA        5444     A
GRA        TYIII   SA3    A5+B1       R2 = R1 + 1;  R2W = [R2]                        GRA        5445     A
GRA                SA4    F.RDT+B2    D = RDT(OC[R1])                                 GRA        5446     A
GRA                LX3    -IH.RFP                                                     GRA        5447     A
GRA                BX5    "RN"X3      R = RF[R2]                                      GRA        5448     A
GRA                SA2    "SB"+X5     LJW = [SB+R]                                    GRA        5449     A
GRA                BX3    X0*X3                                                       GRA        5450     A
GRA                AX2    B3,X2       LJW = SHIFT(LJW,-SC)                            GRA        5451     A
GRA                SX1    X2                                                          GRA        5452     A
GRA                BX7    X1+X3       RF[R2W] = SETX(LJW)                             GRA        5453     A
GRA                LX7    IH.RFP                                                      GRA        5454     A
GRA                EQ     TYIT                                                        GRA        5455     A
GRA        JIN     SPACE  2,12                                                        GRA        5456     A
GRA                PROCESS (JIN,JPBB)                                                 GRA        5457     A
GRA                SB3    I.BRP                                                       GRA        5458     A
GRA                                                                                   GRA        5459          I
          -FCC0909
GRA        JIN1    BX7    "RN"X5      R = RI[R1]                                      GRA        5460          I
          -FCC0909

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA              SA3    A5+1        R2 = R1 + 1                                  FCC0909      43    A
GRA              PL     X3,JIN3     IF [R2] < 0   */ NOT SPECIAL *JIN*           FCC0909      44    A
GRA              SA2    =XL.HB                                                   FCC0909      45    A
GRA              SX6    X2-12                                                    FCC0909      46    A
GRA              PL     X6,JIN2     IF L.HB > 8   */ *HB* IS NON EMPTY           FCC0909      47    A
GRA              SA2    RAT+1                                                    FCC0909      48    A
GRA              MX6    27          I = 27                                       FCC0909      49    A
GRA       JIN1   NZ     X2,JIN2     IF RAT(I) " 0                                FCC0909      50    A
GRA              SA2    A2+B1                                                    FCC0909      51    A
GRA              LX6    1           I = I - 1                                    FCC0909      52    A
GRA              MI     X6,JIN1     IF I " 0                                     FCC0909      53    A
GRA              EQ     JIN3                    */ NO REGISTERS ASSIGNED         FCC0909      54    A
GRA                                                                              FCC0909      55    A
GRA      *       HOLDING BLOCK IS OR WILL BE NON-EMPTY, SETUP CODE TO CHECK      FCC0909      56    A
GRA      *       FOR JUMP TO TOP OF LOOP LABEL BEFORE *JP* INSTRUCTION.          FCC0909      57    A
GRA                                                                              FCC0909      58    A
GRA       JIN2   SA2    LGL                                                      FCC0909      59    A
GRA              SX7    X2                                                       FCC0909      60    A
GRA              TYPEI  STT,B6      SRI( STT  TI,,,SETX(LGL) )                   FCC0909      61    A
GRA              SX6    B6-4                                                     FCC0909      62    A
GRA              SX2    SO.TLOCK+3                                               FCC0909      63    A
GRA              LX2    R1.SOP                                                   FCC0909      64    A
GRA              BX6    X2+X6                                                    FCC0909      65    A
GRA              TYPEI  RS,X6       SRI( RS  TI-4,TLOCK+3 )                      FCC0909      66    A
GRA              BX7    "RN"X5                                                   FCC0909      67    A
GRA              SA3    "SB"+X7                                                  FCC0909      68    A
GRA              SA2    LIH                                                      FCC0909      69    A
GRA              LX3    -I.BRP      I = BR[SB+RI[R1]]                            FCC0909      70    A
GRA              SX7    B6-8        J = TI - 8                                   FCC0909      71    A
GRA              LX7    IH.RFP                                                   FCC0909      72    A
GRA              BX7    X2+X7                                                    FCC0909      73    A
GRA              TYPEI  JPBB,X3     SRI( JPBB  I,J,0,LIH )                       FCC0909      74    A
GRA              SB2    OC.JIN                                                   FCC0909      75    A
GRA                                                                              FCC0909      76    A
GRA       JIN3   BX7    "RN"X5      R = RI[R1]                                   FCC0909      77    A
GRA              SA3    "SB"+X7     LJW = [SB+R]                                 GRA        5461    A
GRA              BX5    X0*X5                                                    GRA        5462    A
GRA              AX3    B3,X3                                                    GRA        5463    A
GRA              SX2    X3          R = SETX( SHIFT(LJW,-SC) )                   GRA        5464    A
GRA              BX6    X2+X5       RI[R1W] = R                                  GRA        5465    A
GRA              EQ     TYIII                                                    GRA        5466    A
GRA       RJXJ   SPACE  3                                                        CCG2         14    A
GRA              PROCESS RJXJ                                                   CCG2         15    A
GRA              SA3    A5+B1       R2 = R1 + 1                                  CCG2         16    A
GRA              SB3    B0          SC = 0                                       CCG2         17    A
GRA              EQ     JIN3                                                     CCG2         18    A
GRA       JPX    SPACE  3,14                                                     GRA        5467          I
         -FCC0764
GRA       JPX    EJECT                                                           FCC0764      19    A
GRA              PROCESS JPX                                                    GRA        5468    A
GRA              SA1    LIH                                                      GRA        5469    A
GRA              SA2    A5+B1       R2 = R1 + 1                                  GRA        5470    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SB3    B0          SC = 0                                          GRA        5471    A
GRA                BX6    X1-X2                                                       GRA        5472    A
GRA                SX7    X6                                                          GRA        5473    A
GRA                NZ     X7,JIN1     IF LIH " IH[R2]  */ NOT LOOP BACK JUMP          GRA        5474         I
          -FCC0909
GRA                NZ     X7,JIN3     IF LIH " IH[R2]   */ NOT LOOP BACK JUMP         FCC0909      78         I
          -CCG24
GRA                NZ     X7,JPX9     IF LIH " IH[R2]  */ NOT THE LOOP BACK JUMP      CCG24         2    A
GRA                SX7    1                                                           CCG0119      19    A
GRA                SA7    LBJF        LBJF = 1                                        CCG0119      20    A
GRA                SA1    TRD                                                         GRA        5475    A
GRA                ZR     X1,JPX0     IF TRD = 0    */ NO TEST REPLACEMENT            GRA        5476    A
GRA                SA4    A5-4                                                        GRA        5477    A
GRA                BX6    X4-X5                                                       GRA        5478    A
GRA                BX7    "RN"X6                                                      GRA        5479    A
GRA                NZ     X7,JPX0     IF RI[R1] " RI[R1-4]                            GRA        5480    A
GRA                SB6    B6-4        TI = TI - 4   */ DELETE LAST INSTRUCTION        GRA        5481    A
GRA                                                                                   GRA        5482    A
GRA      *         INSERT PREFETCHS FOR NEXT ITERATION OF LOOP                        GRA        5483    A
GRA      *         NOTE - NO *IH* IN THE LD SINCE NO STORES INTO CLASS IN LOOP.       GRA        5484    A
GRA                                                                                   GRA        5485    A
GRA       JPX0     SA3    NAA                                                         GRA        5486    A
GRA                ZR     X3,JPX7     IF NAA = 0    */ NO A-ASSIGNMENTS               GRA        5487    A
GRA                SA0    X3-1        I = NAA                                         GRA        5488    A
GRA                                                                                   GRA        5489    A
GRA       JPX1     SA1    ACT+A0      AI = ACT(I)                                     GRA        5490    A
GRA                SA2    RVT+X1                                                      GRA        5491    A
GRA                SB3    X1          XREG = SETX( AI ) */ X-REGNO OF *LD*            GRA        5492    A
GRA                LX1    -CC.INCP    IP = INC[AI]                                    GRA        5493    A
GRA                SX2    X2+4        J = RCT(XREG) + 4  */ R-NUM OF A-REG            GRA        5494         I
          -CCG25
GRA                SA2    RVT-8+B3    J = RVT(XREG-8)  */ R-NUM OF A-REG              CCG25       184    A
GRA                SA3    B4+X1       CAW = RCT(IP)                                   GRA        5495    A
GRA                LX1    CC.INCP+59-CC.IMP                                           GRA        5496    A
GRA                LX3    59-CA.RAP                                                   GRA        5497    A
GRA                PL     X3,JPX3     IF ^RA[CAW]   */ INC NOT IN A REGISTER          GRA        5498    A
GRA                                                                                   GRA        5499    A
GRA                SB2    OC.SLD      OC = OC.SLD                                     GRA        5500    A
GRA                PL     X1,JPX2     IF ^IM[AI]                                      GRA        5501    A
GRA                SB2    OC.SDL      OC = OC.SDL                                     GRA        5502    A
GRA       JPX2     SA3    A3+2        CCW = RCT(IP+2)                                 GRA        5503    A
GRA                MX7    -CC.REG1L                                                   GRA        5504    A
GRA                LX3    -CC.REG1P                                                   GRA        5505    A
GRA                BX7    -X7*X3      REGNO = REG1[CCW]                               GRA        5506    A
GRA                LX2    R1.RJP                                                      GRA        5507    A
GRA                SA4    RVT+X7      K = RVT(REGNO)                                  GRA        5508    A
GRA                MX7    0           R2W = 0                                         GRA        5509    A
GRA                LX4    R1.RKP                                                      GRA        5510    A
GRA                BX6    X2+X4       R1W = TYI(0,J,K,0)                              GRA        5511    A
GRA                EQ     JPX5                                                        GRA        5512    A
GRA                                                                                   GRA        5513    A
GRA      *         INC NOT IN A REGISTER, MUST BE A CON ( *DAA* RESTRICTION ).        GRA        5514    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                                                                                GRA        5515     A
GRA       JPX3     SA3     A3+B1       C2W = RCT(IP+1)                             GRA        5516     A
GRA                LX2     IH.RFP                                                  GRA        5517     A
GRA                PL      X1,JPX4     IF ^IM[AI]                                  GRA        5518     A
GRA                LX3     -IH.CAP-IH.CAL                                          GRA        5519     A
GRA                MX4     IH.CAL                                                  GRA        5520     A
GRA                BX3     -X3*X4      CA[C2W] = -CA[C2W]                          GRA        5521     A
GRA                LX3     IH.CAL+IH.CAP                                           GRA        5522     A
GRA                                                                                GRA        5523     A
GRA       JPX4     SB2     OC.LD       OC = OC.LD                                  GRA        5524     A
GRA                MX6     0           R1W = 0                                     GRA        5525     A
GRA                BX7     X2+X3       R2W = IHW(0,J,CA[C2W],0)                    GRA        5526     A
GRA                                                                                GRA        5527     A
GRA       JPX5     SX5     B6          R = TI                                      GRA        5528     A
GRA                BX6     X6+X5                                                   GRA        5529     A
GRA                RJ      SRI         SRI(  OC  R,... )                           GRA        5530     A
GRA                                                                                GRA        5531     A
GRA                SA2     RVT+B3      J = RVT(XREG)                               GRA        5532     A
GRA                SX4     SO.LOCK+B3                                              GRA        5533     A
GRA                SB2     OC.RS                                                   GRA        5534     A
GRA                LX4     R1.SOP                                                  GRA        5535     A
GRA                BX6     X4+X5                                                   GRA        5536     A
GRA                LX2     IH.RFP                                                  GRA        5537     A
GRA                BX7     X2          R2W = IHW(0,J,0,0)                          GRA        5538     A
GRA                RJ      SRI         SRI(  RS  R,FL+XREG,R2W )                   GRA        5539     A
GRA                SB2     A0                                                      GRA        5540     A
GRA                SA0     A0-B1       I = I - 1                                   GRA        5541     A
GRA                GT      B2,JPX1     IF I > 0                                    GRA        5542     A
GRA                                                                                GRA        5543     A
GRA        JPX7    SA4     TRD                                                     GRA        5544     A
GRA                SA5     A5                                                      GRA        5545     A
GRA                SB3     B0          SC = 0                                      GRA        5546     A
GRA                SX7     B1                                                      GRA        5547          I
          -FCC0985
GRA                UX6     B2,X5                                                   GRA        5548          I
          -FCC0985
GRA                SA7     LBJ         LBJ = 1                                     GRA        5549          I
          -FCC0985
GRA                UX6     B2,X5                                                   FCC0985      64     A
GRA                ZR      X4,JIN1     IF TRD = 0    */ NO TEST REPLACEMENT        GRA        5550          I
          -FCC0909
GRA                ZR      X4,JIN3     IF TRD = 0    */ NO TEST REPLACEMENT        FCC0909      79          I
          -CCG24
GRA                ZR      X4,JPX9     IF TRD = 0    */ NO TEST REPLACEMENT        CCG24         3     A
GRA                                                                                GRA        5551     A
GRA       *        OUTPUT SPECIAL TEST JUMP,  GE  LUL,LCV,LIH                      GRA        5552     A
GRA                                                                                GRA        5553     A
GRA                SA1     LUL+1                                                   GRA        5554     A
GRA                SA2     LCV+1                                                   GRA        5555     A
GRA                SB2     OC.JPBB                                                 GRA        5556     A
GRA                SA1     RVT+X1      I = RVT(LUL(2))                             GRA        5557     A
GRA                SA3     LIH                                                     GRA        5558     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA              SA2    RVT+X2      J = RVT(LCV(2))                          GRA        5559    A
GRA              PL     X4,JPX8     IF TRD > 0    */ F(I) NOT NEGATIVE       GRA        5560    A
GRA              SB3    X1+                                                  GRA        5561    A
GRA              BX1    X2          SWAP( I , J )                            GRA        5562    A
GRA              SX2    B3                                                   GRA        5563    A
GRA                                                                          GRA        5564    A
GRA       JPX8   SA4    F.RDT+B2    D = RDT(OC.JPBB)                         GRA        5565    A
GRA              LX2    IH.RFP                                               GRA        5566    A
GRA              SX7    B1+B1                                                GRA        5567    A
GRA              PX6    B2,X1       R1W = TYII(OC.JPBB,0,0,I)                GRA        5568    A
GRA              BX2    X2+X3                                                GRA        5569    A
GRA              LX7    IH.CAP                                               GRA        5570    A
GRA              BX7    X7+X2       R2W = IHW(0,J,2,LIH)                     GRA        5571    A
GRA              EQ     TYIT                                                 GRA        5572    A
GRA      *       CONVERT ZR,NZ JUMPS TO TEST THE B-VALUES IF AVAILABLE       CCG24         4    A
GRA                                                                          CCG24         5    A
GRA       JPX9   SA4    A5+B1       R2 = R1 + 1                              CCG24         6    A
GRA              MX6    -3                                                   CCG24         7    A
GRA              LX4    -IH.CAP                                              CCG24         8    A
GRA              BX6    -X6*X4                                               CCG24         9    A
GRA              SX7    X6-2                                                 CCG24        10    A
GRA              PL     X7,JIN3     IF JC[R2] " ZR,NZ                        CCG24        11    A
GRA              BX7    "RN"X5                                               CCG24        12    A
GRA              SA2    "SB"+X7     LJW = [SB+RI[R1]]                        CCG24        13    A
GRA              LX2    -I.BRP                                               CCG24        14    A
GRA              SX2    X2                                                   CCG24        15    A
GRA              ZR     X2,JIN3     IF BR[LJW] = 0  */ NO B-REGISTER VALUE   CCG24        16    A
GRA                                                                          CCG0089       6    A
GRA              BX6    X2                                                   CCG0089       7    A
GRA              SB3    8                                                    CCG0089       8    A
GRA              SA6    RVT                                                  CCG0089       9    A
GRA       JPX10  SA3    RVT-1+B3    SEARCH RVT FOR MATCH OF R-NUM            CCG0089      10    A
GRA              BX6    X2-X3                                                CCG0089      11    A
GRA              SB3    B3-B1                                                CCG0089      12    A
GRA              NZ     X6,JPX10                                             CCG0089      13    A
GRA              SA6    RVT                                                  CCG0146       6    A
GRA              ZR     B3,JIN3     IF R-NUM NO LONGER AVAIL                 CCG0089      14    A
GRA      *                           B3 = 0 WHEN GOING TO JIN3               CCG0089      15    A
GRA                                                                          CCG24        17    A
GRA              SB2    OC.JPBB                                              CCG24        18    A
GRA              LX4    IH.CAP                                               CCG24        19    A
GRA              PX6    B2,X2                                                CCG24        20    A
GRA              BX7    X4                                                   CCG24        21    A
GRA              SA4    F.RDT+B2                                             CCG24        22    A
GRA              EQ     TYIT                                                 CCG24        23    A
GRA       LD     SPACE  3,14                                                 GRA        5573    A
GRA              PROCESS (LD,ILD,TLD)                                        GRA        5574          I
         -CCG1
GRA              PROCESS (LD,ILD,TLD,LDC)                                    CCG1        150          I
         -CCG16
GRA              PROCESS (LD,ILD,TLD,LDC,LDV)                                CCG16       240    A
GRA              LX1    59-CA.RAP                                            GRA        5575    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SX6    B6          TR = TI                                      CCG23        126     A
GRA                PL     X1,LD1      IF ^RA[CAW]                                  GRA         5576     A
GRA                LX1    CA.RAP-CA.VCP                                            GRA         5577     A
GRA                MI     X1,PVC      IF VC[CAW]                                   GRA         5578     A
GRA                                                                               GRA         5579     A
GRA                SB2    OC.LD       OC[R1] = OC.LD                               GRA         5580     A
GRA                                                                               FCC1227        5     A
GRA                SA3    A5+B1       R2 = R1 + 1                                  FCC1227        6     A
GRA                SX6    X3                                                       FCC1227        7     A
GRA                NZ     X6,PAC      IF IH[R2] " 0 */ NOT A F.P. ADDR LOAD        FCC1227        8     A
GRA                LX3    -IH.RFP                                                  FCC1227        9     A
GRA                SX6    X3-1                                                     FCC1227       10     A
GRA                NZ     X6,PAC      IF RF[R2] " 1                                FCC1227       11     A
GRA                SB2    OC.STT      OC = OC.STT   */ FORCE A *SA* INST IN *PAC*  FCC1227       12     A
GRA                                                                               FCC1227       13           I
          -FCC1644X
GRA                EQ     PAC                                                      GRA         5581           I
          -FCC1644X
GRA                SA3    A1+2        CCW = CAW + 2                                FCC1644X       1     A
GRA                MX6    -CC.REG1L                                                FCC1644X       2     A
GRA                LX3    -CC.REG1P                                                FCC1644X       3     A
GRA                BX6    -X6*X3                                                   FCC1644X       4     A
GRA                BX3    X6                                                       FCC1644X       5     A
GRA                AX6    3                                                        FCC1644X       6     A
GRA                ZR     X6,PAC      IF REG1[CCW] < 8  */ IN A B-REGISTER         FCC1644X       7     A
GRA                                                                               FCC1644X       8     A
GRA       *        LCM=I ADDRESS IS IN AN X-REGISTER, USE THAT VALUE               FCC1644X       9     A
GRA                                                                               FCC1644X      10     A
GRA                SA3    RVT+X3      R = RVT(REG1[CCW])                           FCC1644X      11     A
GRA                BX6    X3                                                       FCC1644X      12     A
GRA                LX3    I.STP                                                    FCC1644X      13     A
GRA                BX7    X3+X6                                                    FCC1644X      14     A
GRA                SA7    A5+3        [R1+3] = RNI(0,R,0,R)                        FCC1644X      15     A
GRA                EQ     IRA10                                                    FCC1644X      16     A
GRA                                                                               GRA         5582     A
GRA        LD1     SX6    B6+         RI[R1W] = TI                                 GRA         5583           I
          -CCG23
GRA                SA6    A5+3        [R1+3] = RNI(0,0,0,TI)                       GRA         5584           I
          -CCG23
GRA        LD1     SA6    A5+3        [R1+3] = RNI(0,0,0,TR)                       CCG23        127     A
GRA                BX7    X4                                                       CCG23        128     A
GRA                LX7    59-D.FPP                                                 CCG23        129     A
GRA                PL     X7,LD2      IF ^FP[DI]                                   CCG23        130     A
GRA                SA2    =XHO$OPT                                                 CCG23        131     A
GRA                PL     X2,LD2      IF HO$OPT " 2                                CCG23        132     A
GRA                SX7    B1          FLAG = 1                                     CCG23        133     A
GRA                CALL   MPR#        CALL MPR( R1 , FLAG )                        CCG23        134     A
GRA                SA4    A5+2        RESTORE X4, X6                               CCG23        135     A
GRA                SA3    A4+B1                                                    CCG23        136     A
GRA                BX6    X3                                                       CCG23        137     A
GRA                                                                               GRA         5585     A
GRA       *        ADJUST *RF* OF LD,ST,STT INSTRUCTIONS                           GRA         5586     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                                                                    GRA        5587    A
GRA       LD2     SA3    A5+B1       R2 = R1 + 1                       GRA        5588    A
GRA               LX3    -IH.RFP                                       GRA        5589    A
GRA               BX1    "RN"X3      R = RF[R2]                        GRA        5590    A
GRA               SA2    "SB"+X1     LJW = [SB+R]                      GRA        5591    A
GRA               BX7    X0*X3                                         GRA        5592    A
GRA               PL     X2,LD3      IF ^INB[LJW]                      GRA        5593    A
GRA               LX2    -I.BRP      LJW = SHIFT(LJW,-I.BRP)           GRA        5594    A
GRA       LD3     SX2    X2          J = SETX(LJW)                     GRA        5595    A
GRA               SA1    DMASK                                         GRA        5596    A
GRA               BX7    X2+X7                                         GRA        5597    A
GRA               BX5    X0*X5                                         GRA        5598    A
GRA               LX7    IH.RFP      RF[R2W] = J                       GRA        5599    A
GRA               BX6    X5+X6                                         GRA        5600         I
          -CCG23
GRA               BX6    X5+X6       RI[R1W] = TR                      CCG23       138    A
GRA               BX4    -X1*X4      D = [DI] & ^DMASK                 GRA        5601    A
GRA               EQ     TYIT                                          GRA        5602    A
GRA       STT     SPACE  2                                             GRA        5603    A
GRA      *        STT - *IRA* *AC* OR *CON* CANIDATE                   GRA        5604    A
GRA                                                                    GRA        5605    A
GRA       STT0    SX7    B6                                            GRA        5606    A
GRA               SX6    B6                                            GRA        5607    A
GRA               LX7    I.STP                                         GRA        5608    A
GRA               BX7    X6+X7                                         GRA        5609    A
GRA               SA7    A5+3        [R1+3] = RNI(0,TI,0,TI)           GRA        5610    A
GRA               EQ     LD2                                           GRA        5611    A
GRA                                                                    GRA        5612    A
GRA               PROCESS STT                                          GRA        5613    A
GRA               LX1    59-CA.RAP                                     GRA        5614    A
GRA               PL     X1,STT0     IF ^RA[CAW]                       GRA        5615    A
GRA                                                                    GRA        5616    A
GRA               LX1    CA.RAP-CA.IRAP                                GRA        5617    A
GRA               PL     X1,STT1     IF ^IRA[CAW]  */ NOT ASSIGNED IN *ERC*  GRA   5618    A
GRA               BX6    X4                                            GRA        5619    A
GRA               LX6    59-D.PRSP                                     GRA        5620    A
GRA               PL     X6,STT1     IF ^PRS[DI]   */ NOT THE ONE ERC MARKED  GRA  5621    A
GRA               SA2    A5+7        LIS = R1 + 7  */ LINK WORD OF *RS*  GRA        5622    A
GRA               ZR     X2,STT1     IF [LIS] = 0  */ NOT MOVED BY ERC/RS  GRA      5623    A
GRA                                                                    GRA        5624    A
GRA               MX7    0                                             GRA        5625    A
GRA               SA7    A4+B1       [DI+1] = 0                        GRA        5626    A
GRA               EQ     IRA10                                         GRA        5627    A
GRA                                                                    GRA        5628    A
GRA       STT1    LX1    CA.IRAP-CA.CONP                               GRA        5629    A
GRA               PL     X1,PAC      IF ^CON[CAW]                      GRA        5630    A
GRA                                                                    GRA        5631    A
GRA      *        CHANGE *INC* STT TO A *SA* OR *SS*                   GRA        5632    A
GRA                                                                    GRA        5633    A
GRA               SA1    A1+B1       C2W = RCT(J+1)                    GRA        5634    A
GRA               SA3    A5+B1       R2 = R1 + 1                       GRA        5635    A
GRA               BX1    X3-X1                                         GRA        5636    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                LX3    -IH.RFP                                                GRA        5637    A
GRA                SA2    "SB"+X3     LJW = [SB+RF[R2]]                          GRA        5638    A
GRA                LX3    IH.RFP+42-IH.CAP                                       GRA        5639    A
GRA                MI     X2,STT2     IF INB[LJW]                                GRA        5640    A
GRA                                                                              GRA        5641    A
GRA                MI     X3,STT0     IF CA[R2] < 0 */   SRI  X-B DOESNT EXIST    GRA        5642    A
GRA                                                                              GRA        5643    A
GRA        STT2    LX1    59-IH.CAP                                              GRA        5644    A
GRA                SB2    OC.SA       OC = SA                                    GRA        5645    A
GRA                MX7    -CC.REG1L                                              GRA        5646    A
GRA                PL     X1,STT3     IF CA[R2] = CA[C2W]                        GRA        5647    A
GRA                SB2    OC.SS       OC = OC.SS                                 GRA        5648    A
GRA                                                                              GRA        5649    A
GRA        STT3    SA1    A1+B1       CCW = RCT(J+2)                             GRA        5650    A
GRA                LX1    -CC.REG1P                                              GRA        5651    A
GRA                BX7    -X7*X1                                                 GRA        5652    A
GRA                SA3    RVT+X7      LKW = RVT(REG1(CCW])                       GRA        5653    A
GRA                PL     X2,TYI      IF ^INB[LJW]                               GRA        5654    A
GRA                LX2    -I.BRP      LJW = SHIFT(LJW,-I.BRP)                    GRA        5655    A
GRA                EQ     TYI                                                    GRA        5656    A
GRA        ST      EJECT                                                         GRA        5657    A
GRA       *        ST - ADDRESS CANIDATE OR REDEFINTION OF A REGISTER            GRA        5658    A
GRA                                                                              GRA        5659    A
GRA                PROCESS (ST,TST)                                              GRA        5660    A
GRA                LX1    59-CA.KDP                                              GRA        5661         I
          -FCC0977
GRA                MI     X1,IRA10    IF KD[CAW]    */ DEF HAS NO USES           GRA        5662         I
          -FCC0977
GRA                                                                              GRA        5663         I
          -FCC0977
GRA                BX7    "RN"X5      R = RI[R1]                                 GRA        5664         I
          -FCC0977
GRA                LX1    CA.KDP-CA.RAP                                          GRA        5665         I
          -FCC0977
GRA                SA2    "SB"+X7     LJ = SB + R;  LJW = [LJ]                   GRA        5666         I
          -FCC0977
GRA                MI     X1,ST1      IF RA[CAW]                                 GRA        5667         I
          -FCC0977
GRA                BX7    "RN"X5      R = RI[R1]                                 FCC0977      14    A
GRA                SA2    "SB"+X7     LJ = SB + R;  LJW = [LJ]                   FCC0977      15    A
GRA                LX1    59-CA.RAP                                              FCC0977      16    A
GRA                MI     X1,ST0      IF RA[CAW]                                 FCC0977      17    A
GRA                                                                              GRA        5668    A
GRA                LX2    -I.STP                                                 GRA        5669    A
GRA                SX6    X2          RI[R1W] = ST[LJW]                          GRA        5670         I
          -CCG23
GRA                EQ     LD2                                                    GRA        5671         I
          -CCG23
GRA                SX6    X2          TR = ST[LJW]                               CCG23       139    A
GRA                EQ     LD1                                                    CCG23       140    A
GRA                                                                              GRA        5672    A
GRA        ST1     LX1    CA.RAP-CA.VCP                                          GRA        5673         I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
                -FCC0977
GRA      ST0    LX1    CA.RAP-CA.KDP                                   FCC0977      18    A
GRA             PL     X1,ST1       IF ^KD[CAW]                        FCC0977      19    A
GRA             LX4    59-D.INCP                                       FCC0977      20    A
GRA             PL     X4,IRA10     IF ^INC[DI]                        FCC0977      21    A
GRA                                                                    FCC0977      22    A
GRA      *      CLEAR LINK WORD OF INCREMENT SO WE DONT GET GARBAGE IN THE  FCC0977  23    A
GRA      *      *RF* FIELD OF MEM REFS IN THE INCREMENT FIRST CASE.    FCC0977      24    A
GRA                                                                    FCC0977      25    A
GRA             SB6    B6-4         TI = TI - 4   */ BACK OVER INC INSTRUCTION  FCC0977  26    A
GRA             MX6    0                                               FCC0977      27    A
GRA             SA6    A2           [LJ] = 0      */ CLEAR LINK WORD OF *INC*  FCC0977  28    A
GRA             EQ     IRA10                                           FCC0977      29    A
GRA                                                                    FCC0977      30    A
GRA      ST1    LX1    CA.KDP-CA.VCP                                   FCC0977      31    A
GRA             PL     X1,PAC0      IF ^VC[CAW]                        GRA        5674    A
GRA                                                                    GRA        5675    A
GRA      *      PROCESS REDEFINITION OF A REGISTER                     GRA        5676    A
GRA                                                                    GRA        5677    A
GRA             SA0    A2                                              GRA        5678    A
GRA             SA1    A1+2         CCW = RCT(J+2)                     GRA        5679    A
GRA             MX7    -CC.REG1L                                       GRA        5680    A
GRA             LX1    -CC.REG1P                                       GRA        5681    A
GRA             BX5    -X7*X1       REGNO = REG1[CCW]                  GRA        5682    A
GRA             SB2    X2+4                                            GRA        5683         I
                -FCC0414
GRA             NE     B2,B6,ST1C   IF XR[LJW] " TI-4  */ RS WONT FOLLOW REDEF  GRA  5684    I
                -FCC0414
GRA                                                                    GRA        5685         I
                -FCC0414
GRA      *      CHECK *ST* PREDECESSOR                                 GRA        5686         I
                -FCC0414
GRA                                                                    GRA        5687         I
                -FCC0414
GRA             SA3    B7+X2        R1P = TXT(XR[LJW])                 GRA        5688         I
                -FCC0414
GRA             SB3    OC.DEF                                          GRA        5689         I
                -FCC0414
GRA             UX7    B2,X3                                           GRA        5690         I
                -FCC0414
GRA             NE     B2,B3,ST1A   IF OC[R1P] " OC.DEF                GRA        5691         I
                -FCC0414
GRA                                                                    GRA        5692         I
                -FCC0414
GRA      *      MOVED STORE FROM INNER LOOP, OMIT LAST DEF AND SKIP RS GRA        5693         I
                -FCC0414
GRA                                                                    GRA        5694         I
                -FCC0414
GRA             SB6    B6-4         TI = TI - 4   */ OMIT LAST *DEF*  GRA        5695         I
                -FCC0414
GRA             SA3    RVT+X5                                          GRA        5696         I
                -FCC0414

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                 SX7    X3          R = RVT(REGNO)                                  GRA          5697          I
          -FCC0414
GRA                 LX3    I.STP                                                       GRA          5698          I
          -FCC0414
GRA                 BX7    X3+X7       [LJ] = RNI(0,R,0,R)                             GRA          5699          I
          -FCC0414
GRA                 SA7    A0                                                          GRA          5700          I
          -FCC0414
GRA                 EQ     ST4A                                                        GRA          5701          I
          -FCC0414
GRA                                                                                    GRA          5702          I
          -FCC0414
GRA       *         SEE IF *OP* CAN *PRS* TO A B-REG  ( S,SA,SS,STT )                   GRA          5703          I
          -FCC0414
GRA                                                                                    GRA          5704          I
          -FCC0414
GRA        ST1A     SX7    X5-8                                                        GRA          5705          I
          -FCC0414
GRA                 SA3    N.HB                                                        FCC0414        47     A
GRA                 ZR     X3,ST1A     IF N.HB = 0   */ INNER LOOP                     FCC0414        48     A
GRA                                                                                    FCC0414        49     A
GRA       *         OUTER LOOP, CHECK FOR POST STORE OF A REGISTER ASSIGNMENT MADE      FCC0414        50     A
GRA       *         IN AN INNER LOOP AND EXTENDED TO THIS LOOP.                         FCC0414        51     A
GRA                                                                                    FCC0414        52     A
GRA                 SA3    B7+X2       R1P = TXT(XR[LJW])                              FCC0414        53     A
GRA                 SB3    OC.XMT                                                      FCC0414        54     A
GRA                 UX7    B2,X3                                                       FCC0414        55     A
GRA                 EQ     B2,B3,ST0A  IF OC[R1P] = OC.XMT                             FCC0414        56     A
GRA                 SB3    OC.SA                                                       FCC0414        57     A
GRA                 NE     B2,B3,ST0B  IF OC[R1P] " OC.SA                              FCC0414        58     A
GRA                 LX3    -R1.RKP                                                     FCC0414        59     A
GRA                 BX6    "RN"X3                                                      FCC0414        60     A
GRA                 NZ     X6,ST1A     IF RK[R1P] " 0  */ NOT  * SA  RI,RJ *           FCC0414        61     A
GRA        ST0A     LX7    -R1.RJP                                                     FCC0414        62     A
GRA                 BX6    "RN"X7                                                      FCC0414        63     A
GRA                 SA3    B7+X6       R1P = TXT(RJ[R1P])                              FCC0414        64     A
GRA                 UX7    B2,X3                                                       FCC0414        65     A
GRA        ST0B     SB3    OC.DEF                                                      FCC0414        66     A
GRA                 NE     B2,B3,ST1A  IF OC[R1P] " OC.DEF                             FCC0414        67     A
GRA                 LX7    -R1.SOP                                                     FCC0414        68     A
GRA                 MX6    -SO.REGL                                                    FCC0414        69     A
GRA                 BX7    X5-X7                                                       FCC0414        70     A
GRA                 BX6    -X6*X7                                                      FCC0414        71     A
GRA                 ZR     X6,ST4A     IF SOREG[R1P] = REGNO  */ SAME CANIDATES        FCC0414        72          I
          -CCG0128
GRA                 BX6       X3-X2                                                    FCC1568X        5          I
          -CCG0128
GRA                 NZ     X6,ST0C     IF SOREG[R1P] .NE. REGNO */ DIF. CANDIDATES     CCG0128         8     A
GRA                 SA3    A3+2        DIP = R1P + 2                                   CCG0128         9     A
GRA                 LX3    59-D.XUP                                                    CCG0128        10     A
GRA                 MI     X3,ST1C     IF XU[DIP]   */ VAR WAS REDEF IN THIS SEQ.      CCG0128        11     A
GRA                 EQ     ST4A                                                        CCG0128        12     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                                                                    CCG0128      13     A
GRA       ST0C    BX6     X3-X2                                        CCG0128      14     A
GRA               BX7        "RN"X6                                    FCC1568X      6     A
GRA               ZR         X7,ST1C  IF RI[R1P] = XR[LJW] */ REGNO-S " & DEF,ST   FCC1568X      7     A
GRA                                                                    FCC0414      73     A
GRA       *       SEE IF *OP* CAN PRS TO A B-REG ( S,SA,SS,STT )       FCC0414      74     A
GRA                                                                    FCC0414      75     A
GRA       ST1A    SB2     X2+4                                         FCC0414      76     A
GRA               NE      B2,B6,ST1C  IF XR[LJW] " TI-4  */ RS WONT FOLLOW REDEF  FCC0414      77     A
GRA               SA3     B7+X2       R1P = TXT(XR[LJW])               FCC0414      78     A
GRA               SX7     X5-8                                         FCC0414      79     A
GRA               UX6     B2,X3                                        FCC0414      80     A
GRA               PL      X7,ST3      IF REGNO > 7  */ ^ IN A B-REG    GRA        5706     A
GRA               SB3     OC.CLR                                       GRA        5707     A
GRA               NE      B2,B3,ST1B  IF OC[R1P] " OC.CLR              GRA        5708     A
GRA                                                                    GRA        5709     A
GRA       *       CLR PRS TO A B-REG, CHANGE TO A *  S  RI,0 *         GRA        5710     A
GRA                                                                    GRA        5711     A
GRA               SA4     F.RDT+OC.S  DJ = RDT(OC.S)                   GRA        5712     A
GRA               SB2     OC.S                                         GRA        5713     A
GRA               PX7     B2,X3       OC[R1P] = OC.S                   GRA        5714     A
GRA               SX6     B1                                           GRA        5715     A
GRA               SA7     A3                                           GRA        5716     A
GRA               LX6     D.SZP                                        GRA        5717     A
GRA               IX7     X4-X6       SZ[DJ] = 1                       GRA        5718     A
GRA               SA7     A3+2        [R1P+2] = DJ                     GRA        5719     A
GRA               EQ      ST3                                          GRA        5720     A
GRA                                                                    GRA        5721     A
GRA       ST1B    SA4     DUM.JT+B2                                    GRA        5722     A
GRA               SB3     OC.SA                                        GRA        5723     A
GRA               LX4     59-29                                        GRA        5724     A
GRA               PL      X4,ST1C     IF ^BRS(OC[R1P])  */ CANT PRS TO A B-REG  GRA        5725     A
GRA                                                                    GRA        5726     A
GRA       *       OUTPUT *SA* IF PRED IS A *EU* SA PUT OUT BY *PVC* SO AS TO  GRA        5727     A
GRA       *       PRESERVE THE X-REG R-NUMBER                          GRA        5728     A
GRA                                                                    GRA        5729     A
GRA               NE      B2,B3,ST3   IF OC[R1P] " OC.SA               GRA        5730     A
GRA               LX3     -R1.RKP                                      GRA        5731     A
GRA               BX7     "RN"X3                                       GRA        5732     A
GRA               NZ      X7,ST3      IF RK[R1P] " 0  */ NOT  SA RI,RJ GRA        5733     A
GRA                                                                    GRA        5734     A
GRA       *       OUTPUT *SA/XMT* SO THAT *RS* WILL FOLLOW DEFINITION OF RESULT  GRA        5735     A
GRA                                                                    GRA        5736     A
GRA       ST1C    SB2     OC.XMT                                       GRA        5737     A
GRA               SX7     X5-8                                         GRA        5738     A
GRA               BX6     X2+X7                                        GRA        5739     A
GRA               PL      X6,ST2      IF REGNO > 7 & ^INB[LJW]         GRA        5740     A
GRA               SB2     OC.SA                                        GRA        5741     A
GRA               PL      X2,ST2      IF ^INB[LJW]                     GRA        5742     A
GRA               LX2     -I.BRP      LJW = SHIFT(LJW,-I.BRP)          GRA        5743          I
          -CCG0172
GRA               LX6     X2                                           CCG0172       7     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                LX2    -I.BRP       LJW = SHIFT(LJW,-I.BRP)                    CCG0172        8     A
GRA                LX6    59-I.MIBP                                               CCG0172        9     A
GRA                PL     X6,ST2       IF ^MIB(LJW)                               CCG0172       10     A
GRA                SB2    OC.SS                                                   CCG0172       11     A
GRA                SX2    X2           RJ = SETX(LJW)                             CCG0172       12     A
GRA                LX2    R1.RKP                                                  CCG0172       13     A
GRA                EQ     ST2A                   */ TO SRI WITH RJ SWITCHED TO RK CCG0172       14     A
GRA                                                                               CCG0172       15     A
GRA        ST2     SX2    X2           RJ = SETX(LJW)                             GRA          5744     A
GRA                SX6    B6                                                      GRA          5745          I
           -CCG0172
GRA                LX2    R1.RJP                                                  GRA          5746          I
           -CCG0172
GRA                LX2    R1.RJP                                                  CCG0172       16     A
GRA                                                                               CCG0172       17     A
GRA        ST2A    SX6    B6                                                      CCG0172       18     A
GRA                BX6    X2+X6                                                   GRA          5747     A
GRA                MX7    0                                                       GRA          5748     A
GRA                SX2    B6           LJW = TI                                   GRA          5749          I
           -FCC0514
GRA                RJ     SRI          SRI(  XMT/SA  TI,RJ )                      GRA          5750     A
GRA                                                                               GRA          5751     A
GRA        *       OUTPUT *RS* TO FORCE THE RESULT BACK TO THE REGISTER           GRA          5752     A
GRA                                                                               GRA          5753     A
GRA        ST3     SA3    RVT+X5       F = RVT(REGNO)  */ OLD R-NUM IN REG        GRA          5754          I
           -FCC0514
GRA        ST3     SA4    A5+2         DI = R1 + 2                                FCC0514       98     A
GRA                LX4    59-D.XUP                                                FCC0514       99     A
GRA                PL     X4,ST3A      IF ^XU[DI]    */ IF NO EXTENDED USES       FCC0514      100     A
GRA                                                                               FCC0514      101     A
GRA                RJ     APD          ADJUST PREVIOUS DEF TO SAVE OLD VALUE      FCC0514      102     A
GRA                                                                               FCC0514      103     A
GRA        ST3A    SA3    RVT+X5       F = RVT(REGNO)  */ OLD R-NUM IN REGISTER   FCC0514      104     A
GRA                SX4    X5+SO.LOCK                                              GRA          5755     A
GRA                SB2    OC.RS                                                   GRA          5756     A
GRA                SX6    X2           R = SETX(LJW)                              GRA          5757          I
           -FCC0514
GRA                SX6    B6-4         R = TI - 4                                 FCC0514      105     A
GRA                LX4    R1.SOP                                                  GRA          5758     A
GRA                SA6    A3           RVT(REGNO) = R  */ UPDATE R-NUM IN REG     GRA          5759     A
GRA                SB3    X3+2                                                    GRA          5760     A
GRA                BX6    X4+X6                                                   GRA          5761     A
GRA                SA4    B7+B3        DIP = TXT(F+2)  */ CHECK PRED FOR A *CA*   GRA          5762     A
GRA                SX7    A4-1         R2P = DIP - 1                              GRA          5763          I
           -FCC0514
GRA                SX7    A4-B1        R2P = DIP - 1                              FCC0514      106     A
GRA                LX4    59-D.PIP                                                GRA          5764     A
GRA                MI     X4,ST4       IF PI[DIP]                                 GRA          5765     A
GRA                SX7    X7+4         R2P = R2P + 4                              GRA          5766     A
GRA        ST4     SA4    X7                                                      GRA          5767     A
GRA                MX7    -IH.CAIHL                                               GRA          5768     A
GRA                LX3    IH.RFP                                                  GRA          5769     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                BX4    -X7*X4                                          GRA        5770    A
GRA                BX7    X3+X4       R2W = IHW(0,F,CAIH[R2P])            GRA        5771    A
GRA                RJ     SRI         SRI(  RS  R,FL+REGNO,R2W )          GRA        5772    A
GRA                                                                       GRA        5773    A
GRA      *         UPDATE R-NUMBER IN THE LINK WORD OF THE PRED           GRA        5774    A
GRA                                                                       GRA        5775    A
GRA       ST4A     SA1    A1-2                                            GRA        5776    A
GRA                SA3    RVT+X5      RN = RVT(REGNO)                     GRA        5777    A
GRA                SA2    A0+         LJW = [LJ]                          GRA        5778    A
GRA                SX7    X5-8                                            GRA        5779    A
GRA                PL     X7,ST7      IF REGNO > 7  */ X-REG              GRA        5780    A
GRA                                                                       GRA        5781    A
GRA      *         VALUE IS IN A B-REGISTER                               GRA        5782    A
GRA                                                                       GRA        5783    A
GRA                MI     X2,ST5      IF INB[LJW]                         GRA        5784    A
GRA                MX7    1                                               GRA        5785    A
GRA                LX3    I.BRP                                           GRA        5786    A
GRA                LX7    1+I.INBP                                        GRA        5787    A
GRA                BX7    X3+X7                                           GRA        5788    A
GRA                SA7    A0          [LJ] = RNI(1,0,RN,0)                GRA        5789    A
GRA                LX3    -I.BRP                                          GRA        5790    A
GRA                BX2    X7          LJW = [LJ]                          GRA        5791    A
GRA                                                                       GRA        5792    A
GRA       ST5      LX1    59-CA.EUP                                       GRA        5793    A
GRA                MI     X1,ST6      IF EU[CAW]                          GRA        5794    A
GRA                LX1    CA.EUP-CA.PSPP                                  GRA        5795    A
GRA                MI     X1,IRA10    IF PSP[CAW]   */ ST CAN MOVE        GRA        5796    A
GRA                                                                       GRA        5797    A
GRA      *         MOVE RESULT TO AN X-REG                                GRA        5798    A
GRA                                                                       GRA        5799    A
GRA       ST6      SB3    X2                                              GRA        5800    A
GRA                NZ     B3,ST10     IF XR[LJW] " 0*/ PRED IS IN A X-REG GRA        5801    A
GRA                SX6    B6                                              GRA        5802    A
GRA                BX7    X6+X7                                           GRA        5803    A
GRA                LX6    I.STP                                           GRA        5804    A
GRA                BX7    X6+X7                                           GRA        5805    A
GRA                SA7    A0          [LJ] = RNI(1,TI,RN,TI)              GRA        5806    A
GRA                SB2    OC.SA                                           GRA        5807    A
GRA                EQ     ST9                                             GRA        5808    A
GRA                                                                       GRA        5809    A
GRA      *         VALUE IS IN A X-REGISTER                               GRA        5810    A
GRA                                                                       GRA        5811    A
GRA       ST7      SX7    X5-26B                                          GRA        5812    A
GRA                PL     X7,ST10     IF REGNO > 25B  */ RESULT IN A *ST* REG GRA    5813    A
GRA                                                                       GRA        5814    A
GRA                SA4    A0-B1       DJP = LJ - 1                        GRA        5815    A
GRA                SX6    X2          RN = XR[LJW]                        GRA        5816          I
          -FCC1196
GRA                MX7    -D.USESL+1                                      GRA        5817    A
GRA                SA6    A0          [LJ] = RNI(0,0,0,RN)                GRA        5818    A
GRA                LX0    I.STP                                           FCC1196       6    A
GRA                BX6    X0*X2                                           FCC1196       7    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SA6    A0          ST[LJ] = 0                                  FCC1196        8     A
GRA                LX0    -I.STP                                                  FCC1196        9     A
GRA                LX4    -D.USESP-1                                              GRA         5819     A
GRA                BX7    -X7*X4                                                  GRA         5820     A
GRA                NZ     X7,ST8      IF USES[DJP] > 1                            GRA         5821     A
GRA                LX1    59-CA.PSPP                                              GRA         5822     A
GRA                MI     X1,IRA10    IF PSP[CAW]                                 GRA         5823     A
GRA                                                                               GRA         5824     A
GRA        ST8     SX4    B6                                                      GRA         5825     A
GRA                LX4    I.STP                                                   GRA         5826     A
GRA                BX7    X4+X6                                                   GRA         5827     A
GRA                SA7    A0          [LJ] = RNI(0,TI,0,RN)                       GRA         5828          I
           -FCC1196
GRA                SA7    A0          ST[LJ] = TI                                 FCC1196       10     A
GRA                SB2    OC.XMT                                                  GRA         5829     A
GRA                SX3    X7+                                                     GRA         5830     A
GRA                                                                               GRA         5831     A
GRA        ST9     SX6    B6                                                      GRA         5832     A
GRA                LX3    R1.RJP                                                  GRA         5833     A
GRA                MX7    0                                                       GRA         5834     A
GRA                BX6    X3+X6                                                   GRA         5835     A
GRA                RJ     SRI         SRI(  SA/XMT  TI,RN )                       GRA         5836     A
GRA                                                                               GRA         5837     A
GRA       *        OUTPUT STORE IF NOT MOVABLE                                    GRA         5838     A
GRA                                                                               GRA         5839     A
GRA        ST10    SA1    A1                                                      GRA         5840     A
GRA                LX1    59-CA.PSPP                                              GRA         5841     A
GRA                MI     X1,IRA10    IF PSP[CAW]                                 GRA         5842     A
GRA                LX1    CA.PSPP-CA.ACP                                          CCG25        185     A
GRA                MI     X1,IRA10    IF AC[CAW]    */ A/X CANIDATE               CCG25        186     A
GRA                                                                               GRA         5843     A
GRA                SA5    A5                                                      GRA         5844     A
GRA                SA2    A0                                                      GRA         5845     A
GRA                SA4    A5+2                                                    GRA         5846     A
GRA                LX2    -I.STP                                                  GRA         5847     A
GRA                SX6    X2          RI[R1W] = ST[LJW]                           GRA         5848          I
           -CCG23
GRA                SX6    X2          TR = ST[LJW]                                CCG23        141     A
GRA                UX7    B2,X5                                                   GRA         5849     A
GRA                EQ     LD2                                                     GRA         5850          I
           -CCG23
GRA                EQ     LD1                                                     CCG23        142     A
GRA        APD     EJECT                                                          FCC0514      107     A
GRA       **       APD - ADJUST PREVIOUS DEFINITION OF A CANIDATE TO PRESERVE     FCC0514      108     A
GRA       *        PREVIOUS VALUE, BECAUSE USES OF IT EXTEND PAST THE CURRENT     FCC0514      109     A
GRA       *        DEFINITION ( *XU* ).                                           FCC0514      110     A
GRA       *                                                                       FCC0514      111     A
GRA       *        ENTRY  (A0,X5) = LJ, REGNO AS SETUP BY IRA/ST                  FCC0514      112     A
GRA                                                                               FCC0514      113     A
GRA       *        FIND PREVIOUS DEFINITION OF THIS CANIDATE IN *SEQ*             FCC0514      114     A
GRA                                                                               FCC0514      115     A
GRA        APD     ROUTINE                                                        FCC0514      116     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA              SA3    A5+B1       R2 = R1 + 1;  R2P = R2                    FCC0514    117    A
GRA              SB3    4                                                     FCC0514    118    A
GRA              BX7    X3          R2W = [R2]                                FCC0514    119    A
GRA       APD1   SA3    A3-B3       R2P = R2P - 4                             FCC0514    120    A
GRA              BX6    X7-X3                                                 FCC0514    121    A
GRA              NZ     X6,APD1     IF [R2P] " R2W                            FCC0514    122    A
GRA                                                                           FCC0514    123    A
GRA              SA4    A3+B1       DIP = R2P + 1                             FCC0514    124    A
GRA              SA2    A4+B1       LIP = DIP + 1                             FCC0514    125    A
GRA              LX4    59-D.LDP                                              FCC0514    126    A
GRA              MI     X4,APD2     IF LD[DIP]    */ PREVIOUS DEF IS A *LD*   FCC0514    127    A
GRA              SA3    A3-B1       R1P = R2P - 1                             FCC0514    128    A
GRA              BX6    "RN"X3                                                FCC0514    129    A
GRA              SA2    "SB"+X6     LIP = SB + RI[R1P]                        FCC0514    130    A
GRA                                                                           FCC0514    131    A
GRA       *      CHECK LINK WORD TO SEE IF WE MADE A COPY OF THE VALUE        FCC0514    132    A
GRA                                                                           FCC0514    133    A
GRA       APD2   SX6    X5-8                                                  FCC0514    134    A
GRA              PL     X6,APD3     IF REGNO > 8  */ CANIDATE IN AN X-REGISTER FCC0514    135    A
GRA                                                                           FCC0514    136    A
GRA       *      CANIDATE IN B-REG, SEE IF X-REG VALUE IS AVAILABLE           FCC0514    137    A
GRA                                                                           FCC0514    138    A
GRA              SA3    A1-2                                                  FCC0514    139    A
GRA              SB2    OC.SA       OC = OC.SA                                FCC0514    140    A
GRA              LX2    -I.BRP                                                FCC0514    141    A
GRA              SX7    X2          ORV = BR[LIP]                             FCC0514    142    A
GRA              LX3    59-CA.EUP                                             FCC0514    143    A
GRA              LX2    I.BRP                                                 FCC0514    144    A
GRA              PL     X3,APD4     IF ^EU[RCT(J)]  */ NO COPY MADE           FCC0514    145    A
GRA                                                                           FCC0514    146    A
GRA              SX7    X7+2                                                  CCG0128     15    A
GRA              SA3    B7+X7       DI = TXT(ORV+2) */ OF DEF                 CCG0128     16    A
GRA              SX7    B1                                                    CCG0128     17    A
GRA              LX7    D.XUP                                                 CCG0128     18    A
GRA              BX7    X3+X7       XU[DI] = 1    */ INDICATE EXT USE TO IRA/ST CCG0128     19    A
GRA              SA7    A3                                                    CCG0128     20    A
GRA              MX7    1                                                     FCC0514    147    A
GRA              BX6    -X7*X2      INB[LIP] = 0  */ MAKE OLD B-VAL UNAVAILABLE FCC0514    148    A
GRA              SA6    A2                                                    FCC0514    149    A
GRA              EQ     APD                                                   FCC0514    150    A
GRA                                                                           FCC0514    151    A
GRA       APD3   SB2    OC.XMT      OC = OC.XMT                               FCC0514    152    A
GRA              SX7    X2          ORV = XR[LIP] */ OLD R-NUM VALUE          FCC0514    153    A
GRA                                                                           FCC0514    154    A
GRA       *      NO COPY OF CANIDATE AVAILABLE, INSERT *XMT/SA* TO SAVE OLD   FCC0514    155    A
GRA       *      VALUE OF PREVIOUS DEF BEFORE LAST INSTRUCTION IN *TXT*.      FCC0514    156    A
GRA                                                                           FCC0514    157    A
GRA       APD4   SX6    B6-B3       R = TI - 4                                FCC0514    158    A
GRA              BX4    X6                                                    FCC0514    159    A
GRA              LX6    I.STP                                                 FCC0514    160    A
GRA              BX6    X4+X6       [LIP] = RNI(0,R,0,R)                      FCC0514    161    A
GRA              SA6    A2                                                    FCC0514    162    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SA3    B7+X4       R1W = TXT(R)                                FCC0514     163    A
GRA                SX2    B3                                                      FCC0514     164    A
GRA                IX6    X3+X2       RI[R1W] = RI[R1W] + 4                       FCC0514     165    A
GRA                LX7    R1.RJP                                                  FCC0514     166    A
GRA                SA6    B7+B6       TXT(TI) = R1W  */ MOVE LAST INST UP         FCC0514     167    A
GRA                PX7    B2,X7                                                   FCC0514     168    A
GRA                BX7    X4+X7                                                   FCC0514     169    A
GRA                SA7    A3          TXT(R) = TYI(OC,ORV,0,R)*/ INSERT XMT/SA    FCC0514     170    A
GRA                SA3    A3+B1                                                   FCC0514     171    A
GRA                BX6    X3          TXT(TI+1) = TXT(R+1)                        FCC0514     172    A
GRA                SA6    A6+B1                                                   FCC0514     173    A
GRA                MX7    0                                                       FCC0514     174    A
GRA                SA7    A3          TXT(R+1) = 0                                FCC0514     175    A
GRA                SA3    A3+B1                                                   FCC0514     176    A
GRA                BX6    X3          TXT(TI+2) = TXT(R+2)                        FCC0514     177    A
GRA                SA6    A6+B1                                                   FCC0514     178    A
GRA                SA7    A6+B1       TXT(TI+3) = 0                               FCC0514     179    A
GRA                SA2    F.RDT+B2                                                FCC0514     180    A
GRA                BX7    X2          TXT(R+2) = RDT(OC)                          FCC0514     181    A
GRA                SA7    A3                                                      FCC0514     182    A
GRA                                                                               FCC0514     183    A
GRA       *        ADJUST THE LINK WORD OF THE STORE PREDECESSOR                  FCC0514     184    A
GRA                                                                               FCC0514     185    A
GRA                SA2    A0          LJW = [LJ]                                  FCC0514     186    A
GRA                BX4    X2          OLW = LJW     */ SAVE ORIGINAL LW           FCC0514     187    A
GRA                SB2    36          SC = 36                                     FCC0514     188    A
GRA                SX7    B3                                                      FCC0514     189    A
GRA                SB6    B6-B3       TI = TI - 4                                 FCC0514     190    A
GRA       APD5     AX3    B2,X2                                                   FCC0514     191    A
GRA                SB3    X3                                                      FCC0514     192    A
GRA                NE     B3,B6,APD6  IF RN[SHIFT(LJW,-SC)] " TI                  FCC0514     193    A
GRA                LX6    B2,X7                                                   FCC0514     194    A
GRA                IX2    X6+X2       LJW = LJW + SHIFT(4,SC)                     FCC0514     195    A
GRA       APD6     SB2    B2-18       SC = SC - 18                                FCC0514     196    A
GRA                PL     B2,APD5     IF SC \ 0                                   FCC0514     197    A
GRA                                                                               FCC0514     198    A
GRA                BX6    X2                                                      FCC0514     199    A
GRA                SA6    A2          [LJ] = LJW                                  FCC0514     200    A
GRA                SB6    B6+8        TI = TI + 8                                 FCC0514     201    A
GRA                                                                               FCC0514     202    A
GRA                SA3    A0-3        R1P = LJ - 3  */ CHECK ST PRED              FCC0514     203    A
GRA                SB3    OC.XMT                                                  FCC0514     204    A
GRA                UX7    B2,X3                                                   FCC0514     205    A
GRA                NE     B2,B3,APD   IF OC[R1P] " OC.XMT                         FCC0514     206    A
GRA                LX3    -R1.RJP                                                 FCC0514     207    A
GRA                BX7    "RN"X3      R = RJ[R1P]                                 FCC0514     208    A
GRA                SA3    "SB"+X7     LJP = SB + R  */ PRED OF *XMT*              FCC0514     209    A
GRA                BX7    X4-X3                                                   FCC0514     210    A
GRA                NZ     X7,APD      IF [LJP] " OLW  */ NOT OMITTED *XMT*        FCC0514     211    A
GRA                SA6    A3+         [LJP] = LJW   */ UPDATE LINK WORD OF PRED   FCC0514     212    A
GRA                EQ     APD                                                     FCC0514     213    A
GRA       PVC      EJECT                                                           GRA        5851    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA      **        PVC - PROCESS VALUE CANIDATE DEFINITION ( LD OR CON )          GRA        5852     A
GRA                                                                               GRA        5853     A
GRA       PVC      SA1    A1                                                      GRA        5854     A
GRA                LX1    59-CA.KDP                                               GRA        5855     A
GRA                PL     X1,PVC1     IF ^KD[CAW]                                 GRA        5856     A
GRA                                                                               GRA        5857     A
GRA                MX6    0                                                       GRA        5858     A
GRA                SA6    A4+B1       [DI+1] = 0                                  GRA        5859     A
GRA                EQ     IRA10                                                   GRA        5860     A
GRA                                                                               GRA        5861     A
GRA       PVC1     SA2    A1+2        CCW = RCT(J+2)                              GRA        5862     A
GRA                MX7    -CC.REG1L                                               GRA        5863     A
GRA                LX2    -CC.REG1P                                               GRA        5864     A
GRA                BX4    -X7*X2      REGNO = REG1[CCW]                           GRA        5865     A
GRA                SA3    RVT+X4      R = RVT(REGNO)                              GRA        5866     A
GRA                SX6    X4-8                                                    GRA        5867     A
GRA                PL     X6,PVC5     IF REGNO > 7                                GRA        5868     A
GRA                                                                               GRA        5869     A
GRA      *         CANIDATE IN A B-REGISTER                                      GRA        5870     A
GRA                                                                               GRA        5871     A
GRA                LX1    1+CA.KDP                                                GRA        5872     A
GRA                SX7    B6                                                      GRA        5873     A
GRA                LX3    I.BRP                                                   GRA        5874     A
GRA                BX6    X3+X7                                                   GRA        5875     A
GRA                LX7    I.STP                                                   GRA        5876     A
GRA                BX6    X6+X7                                                   GRA        5877     A
GRA                MX7    1                                                       GRA        5878     A
GRA                LX7    1+I.INBP                                                GRA        5879     A
GRA                BX6    X7+X6                                                   GRA        5880     A
GRA                SA6    A5+3        [R1+3] = RNI(1,TI,R,TI)                     GRA        5881     A
GRA                LX1    59-CA.EUP                                               GRA        5882          I
          -CCG0034
GRA                PL     X1,IRA10    IF ^EU[CAW]                                 GRA        5883          I
          -CCG0034
GRA                                                                               GRA        5884          I
          -CCG0034
GRA                LX1    CA.EUP-CA.CONP                                          GRA        5885          I
          -CCG0034
GRA                LX1    59-CA.CONP                                              CCG0034      24     A
GRA                MX7    0                                                       GRA        5886     A
GRA                LX3    R1.RJP-I.BRP                                            GRA        5887     A
GRA                SB2    OC.SA       OC = OC.SA                                  GRA        5888     A
GRA                PL     X1,PVC2     IF ^CON[CAW]                                GRA        5889     A
GRA                                                                               GRA        5890     A
GRA                SA1    A1+B1       C2W = RCT(J+1)                              GRA        5891     A
GRA                LX5    IH.CAP-R1.INP                                           GRA        5892     A
GRA                BX1    X5-X1                                                   GRA        5893     A
GRA                LX1    59-IH.CAP                                               GRA        5894     A
GRA                PL     X1,PVC2     IF CA[C2W] = IN[R1]                         GRA        5895     A
GRA                SX1    B1                                                      CCG0034      25     A
GRA                LX1    I.MIBP                                                  CCG0034      26     A
GRA                BX6    X6+X1                                                   CCG0034      27     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                 SA6    A6          [R1+3] = RNI(3,TI,R,TI)                          CCG0034      28     A
GRA                 SB2    OC.SS                                                        GRA        5896     A
GRA                 LX3    R1.RKP-R1.RJP                                                GRA        5897     A
GRA                 SX1    B6                                                           CCGA062       5          I
          -CCG0034
GRA                 SX6    B6                                                           CCGA062       6          I
          -CCG0034
GRA                 LX1    I.STP                                                        CCGA062       7          I
          -CCG0034
GRA                 BX6    X1+X6                                                        CCGA062       8          I
          -CCG0034
GRA                 SA6    A5+3        [R1+3] = RUE(0,TI,0,TI)                          CCGA062       9          I
          -CCG0034
GRA                                                                                     CCGA062      10     A
GRA       PVC2      SX6    B6                                                           GRA        5898     A
GRA                 BX6    X6+X3                                                        GRA        5899     A
GRA                 RJ     SRI         SRI(  SA  TI,R  /  SS  TI,,R )                   GRA        5900     A
GRA                 EQ     IRA10                                                        GRA        5901     A
GRA                                                                                     GRA        5902     A
GRA       *         CANIDATE IN A X-REGISTER                                             GRA        5903     A
GRA                                                                                     GRA        5904     A
GRA       PVC5      SA2    A4                                                           GRA        5905     A
GRA                 SX6    X4-26B                                                       GRA        5906     A
GRA                 BX5    X3          RST = R       */ R-NUM IN A ST REG               GRA        5907     A
GRA                 LX2    59-D.PSP                                          002270     FCC2165       1     A
GRA                 PL     X6,PVC6     IF REGNO > 25B  */ X6,X7                         GRA        5908     A
GRA                 LX2    59-D.PSP                                                     GRA        5909          I
          -FCC2165
GRA                 SX5    B0          RST = 0                                          GRA        5910     A
GRA                 PL     X2,PVC6     IF ^PS[DI]                                       GRA        5911     A
GRA                                                                                     GRA        5912     A
GRA                 LX3    R1.RJP                                                       GRA        5913     A
GRA                 SX5    B6          RST = TI                                         GRA        5914     A
GRA                 MX7    0                                                            GRA        5915     A
GRA                 BX6    X3+X5                                                        GRA        5916     A
GRA                 LX3    -R1.RJP                                                      GRA        5917     A
GRA                 TYPEI  XMT,X6      SRI(  XMT  TI,R )                                GRA        5918     A
GRA                                                                                     GRA        5919     A
GRA       PVC6      LX5    I.STP                                                        GRA        5920     A
GRA                 BX6    X5+X3                                                        GRA        5921     A
GRA                 SA6    A5+3        [R1+3] = RNI(0,RST,0,R)                          GRA        5922     A
GRA                 LX2    D.PSP-D.PRSP                                                 FCC0812       6     A
GRA                 PL     X2,IRA10    IF ^PRS[DI]                                      FCC0812       7     A
GRA                                                                                     FCC0812       8     A
GRA                 SA4    A5+7        LIRS = R1 + 7;  I = CO[LIRS]  */ RS LINK WD      FCC0812       9     A
GRA                 SA1    B4+X4       CAW = RCT(I)                                     FCC0812      10     A
GRA                 LX1    59-CA.IRAP                                                   FCC0812      11     A
GRA                 MI     X1,IRA10    IF IRA[CAW]   */ RS IS TO BE MOVED OUT           FCC0812      12     A
GRA                                                                                     FCC0812      13     A
GRA                 MX7    0                                                            FCC0812      14     A
GRA                 TYPEI  XMT,B6,X3   SRI(  XMT  TI,R )  */ OUTPUT XMT FOR *RS*        FCC0812      15     A
GRA                 EQ     IRA10                                                        GRA        5923     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA        PAC      SPACE  3,24                                                GRA        5924     A
GRA       **        PAC - PROCESS ADDRESS CANIDATE, MIT = 1,2,3                GRA        5925          I
          -FCC0646
GRA       **        PAC - PROCESS ADDRESS CANIDATE, MIT = 1 - 4                FCC0646      22     A
GRA       *                                                                    GRA        5926     A
GRA       *         ENTRY  (B2) = OC.LD, OC.ST, OC.STT                         GRA        5927     A
GRA                                                                            GRA        5928     A
GRA        PAC0     SB2    OC.ST                                               GRA        5929     A
GRA                                                                            GRA        5930     A
GRA        PAC      SA1    A1+2        CCW = RCT(J+2)                          GRA        5931     A
GRA                 SB2    B2-OC.LD    MODE = OC[R1] - OC.LD                   GRA        5932     A
GRA                 MX7    -CC.REG1L                                           GRA        5933     A
GRA                 LX1    -CC.REG1P                                           GRA        5934     A
GRA                 BX2    -X7*X1      REGN1 = REG1[CCW]                       GRA        5935     A
GRA                 LX1    CC.REG1P-CC.MITP                                    GRA        5936     A
GRA                 BX3    -X7*X1      IT = MIT[CCW]                           GRA        5937     A
GRA                 SA0    X3                                                  GRA        5938     A
GRA                 LX1    CC.MITP-CC.REG2P                                    GRA        5939     A
GRA                 BX7    -X7*X1      REGN2 = REG2[CCW]                       GRA        5940     A
GRA                 SA4    PACA+B2     OCW = PACA(MODE)                        GRA        5941     A
GRA                 LX3    4                                                   GRA        5942          I
          -FCC0646
GRA                 MX6    -12                                                 GRA        5943          I
          -FCC0646
GRA                 LX3    3                                                   FCC0646      23     A
GRA                 MX6    -8                                                  FCC0646      24     A
GRA                 BX3    -X6*X3      IT = IT & 37B                           FCC0977      32     A
GRA                 SB3    X3                                                  GRA        5944     A
GRA                 LX1    B3,X4                                               GRA        5945     A
GRA                 BX6    -X6*X1      NOC = SHIFT(OCW,16*IT) & 7777B          GRA        5946          I
          -FCC0646
GRA                 BX6    -X6*X1      NOC = SHIFT(OCW,8*IT) & 377B            FCC0646      25     A
GRA                 SB3    X6                                                  GRA        5947     A
GRA                 SX6    B6+         R = TI                                  GRA        5948     A
GRA                 SA6    A5+3        [R1+3] = R                              GRA        5949     A
GRA                                                                            GRA        5950     A
GRA                 ZR     B2,PAC1     IF MODE = 0   */ LD                     GRA        5951     A
GRA                 NE     B2,B1,PAC3  IF MODE " 1   */ NOT ST                 GRA        5952     A
GRA                 BX5    "RN"X5                                              GRA        5953     A
GRA                 SA3    "SB"+X5     LIW = [SB+RI[R1]]                       GRA        5954     A
GRA                 LX3    -I.STP                                              GRA        5955     A
GRA                 SX6    X3          R = ST[LIW]                             GRA        5956     A
GRA                                                                            GRA        5957     A
GRA       *         REFORM REFERENCE AS A TYPE I INSTRUCTION                   GRA        5958     A
GRA                                                                            GRA        5959     A
GRA        PAC1     SA3    RVT+X7                                              GRA        5960          I
          -FCC0646
GRA                 SA2    RVT+X2                                              GRA        5961          I
          -FCC0646
GRA        PAC1     SX4    A0-4                                                FCC0646      26     A
GRA                 ZR     X4,PAC4     IF IT = 4                               FCC0646      27     A
GRA                                                                            FCC0646      28     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA       *          IT = 1 - 3 , RJ,RK ARE FROM *RVT*                          FCC0646      29     A
GRA                                                                             FCC0646      30     A
GRA                  SA3    RVT+X7      K = RVT(REGN2)                           FCC0646      31     A
GRA                  SA2    RVT+X2      J = RVT(REGN1)                           FCC0646      32     A
GRA                  SA4    A5+B1       R2 = R1 + 1                              GRA        5962     A
GRA                  LX3    R1.RKP                                               GRA        5963     A
GRA                  BX6    X3+X6                                                GRA        5964     A
GRA                  LX2    R1.RJP                                               GRA        5965     A
GRA                  BX6    X2+X6                                                GRA        5966     A
GRA                  LX4    -IH.RFP                                              GRA        5967     A
GRA                  SX1    B1          ZB = 1        */ ZP BIT                  GRA        5968     A
GRA                  BX7    "RN"X4      F = RF[R2]                               GRA        5969     A
GRA                  SA3    "SB"+X7     LFW = [SB+F]                             GRA        5970     A
GRA                  ZR     X7,PAC2     IF F = 0      */ NO RF OR A0             GRA        5971     A
GRA                                                                             GRA        5972     A
GRA                  SX1    B0          ZB = 0                                   GRA        5973     A
GRA                  PL     X3,PAC2     IF ^INB[LFW]                             GRA        5974     A
GRA                  LX2    -R1.RJP                                              GRA        5975     A
GRA                  SX3    X2          LFW = RVT(REGN1)  */ USE *RF* FROM RVT   GRA        5976     A
GRA                                                                             GRA        5977     A
GRA        PAC2      BX4    X0*X4                                                GRA        5978     A
GRA                  SX3    X3                                                   GRA        5979     A
GRA                  LX1    D.ZPP                                                GRA        5980     A
GRA                  BX7    X3+X4       RF[R2W] = SETX(LFW)                      GRA        5981     A
GRA                  SA2    F.RDT+B3    D = RDT(NOC)                             GRA        5982     A
GRA                  LX7    IH.RFP                                               GRA        5983     A
GRA                  BX4    X2+X1       ZP[D] = ZB                               GRA        5984          I
           -FCC0977
GRA                  PX6    B3,X6                                                GRA        5985          I
           -FCC0646  -FCC0977
GRA                  PX6    B3,X6       R1W = TYI(NOC,J,K,R)                     FCC0646      33          I
           -FCC0977
GRA                  SX3    A0                                                   FCC0977      33     A
GRA                  BX4    X2+X1       ZP[D] = ZB                               FCC0977      34     A
GRA                  AX3    5                                                    FCC0977      35     A
GRA                  PX6    B3,X6       R1W = TYI(NOC,J,K,R)                     FCC0977      36     A
GRA                  LX3    IH.SRFP                                              FCC0977      37     A
GRA                  BX7    X3+X7       SRF[R2W] = IT & 40B                      FCC0977      38     A
GRA                  EQ     TYIT                                                 GRA        5986     A
GRA                                                                             GRA        5987     A
GRA       *          STT - SETUP LINK WORD AND CHECK FOR SPECIAL CASE            GRA        5988     A
GRA                                                                             GRA        5989     A
GRA        PAC3      LX6    I.STP                                                GRA        5990     A
GRA                  SX4    B6                                                   GRA        5991     A
GRA                  BX6    X4+X6                                                GRA        5992     A
GRA                  SA6    A6          [R1+3] = RNI(0,TI,0,TI)                  GRA        5993     A
GRA                  SX1    A0-B1                                                GRA        5994     A
GRA                  SX6    B6          R = TI                                   GRA        5995     A
GRA                  NZ     X1,PAC1     IF IT " 1     */ NOT SPECIAL CASE        GRA        5996     A
GRA                                                                             GRA        5997     A
GRA       *          IT = 1, PARTIALLY MODIFIED *STT* FROM *PSC*, *CA* IS FROM   GRA        5998     A
GRA       *          *RCT* AND *RF* IS RVT(REGN1)                                GRA        5999     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                                                                                  GRA        6000     A
GRA                 SA1    A1-B1       C2W = RCT(J+1)                                GRA        6001     A
GRA                 SA2    RVT+X2                                                    GRA        6002     A
GRA                 MX3    -IH.CAL                                                   GRA        6003     A
GRA                 PX6    B3,X6       R1W = TYII(OC.STT,0,0,TI)                     GRA        6004     A
GRA                 LX3    IH.CAP                                                    GRA        6005     A
GRA                 SA4    F.RDT+B3    D = RDT(NOC)                                  GRA        6006     A
GRA                 BX3    -X3*X1                                                    GRA        6007     A
GRA                 LX2    IH.RFP                                                    GRA        6008     A
GRA                 BX7    X2+X3       R2W = IHW(0,RVT(REGN1),CA[C2W],0)             GRA        6009     A
GRA                 EQ     TYIT                                                      GRA        6010     A
GRA                                                                                  FCC0646      34     A
GRA       *         IT = 4 , GET RJ FROM LINK WORK OF RF IN *SEQ*.                    FCC0646      35     A
GRA                                                                                  FCC0646      36     A
GRA        PAC4     SA4    A5+B1       R2 = R1 + 1                                   FCC0646      37     A
GRA                 LX4    -IH.RFP                                                   FCC0646      38     A
GRA                 BX5    "RN"X4      F = RF[R2]                                    FCC0646      39     A
GRA                 BX4    X0*X4                                                     FCC0646      40     A
GRA                 NZ     X5,PAC5     IF F " 0                                      FCC0646      41     A
GRA                                                                                  FCC0646      42     A
GRA                 SA2    RVT+X2      J = RVT(REGN1)                                FCC0646      43     A
GRA                 MX3    0           K = 0                                         FCC0646      44     A
GRA                 SX1    B1          ZB = 1                                        FCC0646      45     A
GRA                 EQ     PAC8                                                      FCC0646      46     A
GRA                                                                                  FCC0646      47     A
GRA        PAC5     SA3    "SB"+X5     LFW = [SB+F]                                  FCC0646      48     A
GRA                 PL     X3,PAC6     IF ^INB[LFW]                                  FCC0646      49     A
GRA                 LX3    -I.BRP      LFW = SHIFT(LFW,-I.BRP)                       FCC0646      50     A
GRA        PAC6     SX5    X3          F = SETX(LFW)                                 FCC0646      51     A
GRA                 SA3    RVT+X7      K = RVT(REGN2)                                FCC0646      52     A
GRA                 SX1    B0          ZB = 0                                        FCC0646      53     A
GRA                 NZ     X7,PAC7     IF REGN2 " 0  */ RF IS IN *RCT*               FCC0646      54     A
GRA                 SA3    RVT+X2      K = RVT(REGN1)                                FCC0646      55     A
GRA        PAC7     SX2    X5+         J = F                                         FCC0646      56     A
GRA                                                                                  FCC0646      57     A
GRA        PAC8     BX7    X4+X5       RF[R2W] = F                                   FCC0646      58     A
GRA                 SA4    F.RDT+B3    D = RDT(NOC)                                  FCC0646      59     A
GRA                 LX7    IH.RFP                                                    FCC0646      60     A
GRA                 LX2    R1.RJP                                                    FCC0646      61     A
GRA                 LX3    R1.RKP                                                    FCC0646      62     A
GRA                 BX2    X2+X3                                                     FCC0646      63     A
GRA                 BX5    X2+X6                                                     FCC0646      64     A
GRA                 PX6    B3,X5       R1W = TYI(NOC,J,K,R)                          FCC0646      65     A
GRA                 LX1    D.ZPP                                                     FCC0646      66     A
GRA                 BX4    X1+X4       ZP[D] = ZB                                    FCC0646      67     A
GRA                 EQ     TYIT                                                      FCC0646      68     A
GRA                                                                                  GRA        6011     A
GRA        OPT      MACRO  A,B,C                                                     GRA        6012     A
GRA                 VFD    16/OC.A,16/OC.B,16/OC.C,12/                               GRA        6013          I
           -FCC0646
GRA                 VFD    8/OC.A,8/OC.B,8/OC.C,8/OC.B,28/0                          FCC0646      69     A
GRA                 ENDM                                                             GRA        6014     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                                                                    GRA        6015    A
GRA       PACA     BSS    0                                            GRA        6016    A
GRA                OPT    PLD,SLD,SDL                                  GRA        6017    A
GRA                OPT    PST,SST,SDS                                  GRA        6018    A
GRA                OPT    STT,SA,SS                                    GRA        6019    A
GRA       CUC      TITLE  CUC - COLLECT USES COUNTS, SET PRECEDENCE BITS  GRA      6020    A
GRA      **        CUC - COLLECT USES COUNTS, SET PRECEDENCE BITS AND REMOVE  GRA  6021    A
GRA      *         DEAD ( USELESS ) DEFINITIONS FROM A SEQUENCE IN *TXT*.  GRA     6022    A
GRA      *                                                              GRA        6023    A
GRA      *         ENTRY  *TXT* - RLIST INSTRUCTION SEQUENCE WITH R-NUMBERS IN  GRA  6024    A
GRA      *                CANONICAL FORM.                               GRA        6025    A
GRA      *                                                              GRA        6026    A
GRA      *         EXIT   USES, PS,PRS, RJRS FIELDS ADJUSTED AND USELESS  GRA     6027    A
GRA      *                DEFINITIONS REMOVED FROM THE SEQUENCE.        GRA        6028    A
GRA      *                                                              GRA        6029    A
GRA      *         *CUC* MAY BE CALLED AFTER A SEQUENCE HAS BEEN MODIFIED BY  GRA  6030    A
GRA      *         DELETION OR SUBSTITUTION IN SUCH A WAY AS NOT TO INTRODUCE  GRA  6031    A
GRA      *         REDUNDANT INSTRUCTIONS.                              GRA        6032    A
GRA      *         NOTE THAT THE ZP, FP, L2 AND SZ BITS ARE NOT ADJUSTED HERE,  GRA  6033    A
GRA      *         AND MUST BE PRESERVED BY PREVIOUS PROCESSORS.        GRA        6034    A
GRA      *                                                              GRA        6035    A
GRA      *         *CUC* CONSISTS OF A BACKWARDS SCAN TO COLLECT USES AND  GRA    6036    A
GRA      *         SET PRECEDENCE BITS, OPTIONALLY FOLLOWED BY A FOWARD SCAN TO  GRA  6037    A
GRA      *         COMPRESS THE SEQUENCE IF DEAD DEFINITIONS WERE ENCOUNTERED.  GRA  6038    A
GRA                                                                    GRA        6039    A
GRA                ENTRY  CUC                                          GRA        6040         I
         -CCGNC
GRA                QUAL   CUC                                          GRA        6041    A
GRA                                                                    GRA        6042    A
GRA       CUCA     BFMW   D,(^D,RJRS,PI,USES)                          GRA        6043    A
GRA                                                                    GRA        6044    A
GRA       CUC      ROUTINE                                             GRA        6045    A
GRA                SA2    O.TXT                                        GRA        6046    A
GRA                SA3    L.TXT                                        GRA        6047    A
GRA                SA1    CUCA                                         GRA        6048    A
GRA                SB2    B1+B1                                        GRA        6049    A
GRA                S"TB"  X2+B2       TB = O.TXT + 2                   GRA        6050    A
GRA                SB6    D.PSP                                        GRA        6051    A
GRA                MX0    -R1.RIL+1                                    GRA        6052         I
         -CCG11
GRA                SX0    RN.MASK                                      CCG11       122    A
GRA                SA0    B0          DD = 0        */ ADDRESS OF FIRST DEAD DEF  GRA  6053    A
GRA                LX0    1                                            GRA        6054         I
         -CCG11
GRA                IX6    X2+X3                                        GRA        6055    A
GRA                SA5    X6-4        R1 = O.TXT + L.TXT - 4           GRA        6056    A
GRA                MX7    0           [DI+1] = 0                       GRA        6057    A
GRA                SA7    "TB"+B1     [TB+1] = 0    */ CLEAR LINK WD OF *BOS*  GRA  6058    A
GRA                EQ     CUC1                                         GRA        6059    A
GRA       TYI      SPACE  3,14                                         GRA        6060    A
GRA                PROCESS (PST,SST,SDS)                               GRA        6061    A
GRA                BX4    "RN"X5                                       GRA        6062    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                SA3    "TB"+X4     DIP = TB + RI[R1]                          GRA        6063     A
GRA                LX6    B6,X7                                                  GRA        6064     A
GRA                IX2    X7+X3       USES[DIP] = USES[DIP] + 1                  GRA        6065     A
GRA                BX6    X2+X6       PS[DIP] = 1                                GRA        6066     A
GRA                SA6    A3+                                                    GRA        6067     A
GRA                                                                              GRA        6068     A
GRA      *         TYPE I INSTRUCTION PROCESSING                                 GRA        6069     A
GRA                                                                              GRA        6070     A
GRA       TYI      LX5    -R1.RJP                                                GRA        6071     A
GRA                BX6    "RN"X5                                                 GRA        6072     A
GRA                SA3    "TB"+X6     DJ = TB + RJ[R1]                           GRA        6073     A
GRA                LX5    R1.RJP-R1.RKP                                          GRA        6074     A
GRA                IX6    X3+X7       USES[DJ] = USES[DJ] + 1                    GRA        6075     A
GRA                SA6    A3+                                                    GRA        6076     A
GRA                BX5    "RN"X5                                                 GRA        6077     A
GRA                SA2    "TB"+X5     DK = TB + R5[R1]                           GRA        6078     A
GRA                IX6    X2+X7       USES[DK] = USES[DK] + 1                    GRA        6079     A
GRA                SA6    A2+                                                    GRA        6080     A
GRA                                                                              GRA        6081     A
GRA      *         ADVANCE TO NEXT INSTRUCTION, CHECK FOR USELESS DEFINITIONS    GRA        6082     A
GRA                                                                              GRA        6083     A
GRA                PROCESS (LAB,ENT,UJP,RJ3,RJ6,S,FMA,CLR)                       GRA        6084     A
GRA                                                                              GRA        6085     A
GRA       CUC1     SA4    A5-B2       DI = R1 - 2                                GRA        6086     A
GRA                SA5    A4-B2       R1 = DI - 2                                GRA        6087     A
GRA                BX6    X1*X4                                                  GRA        6088     A
GRA                UX2    B3,X5                                                  GRA        6089     A
GRA                SA3    CUC.JT+B3                                              GRA        6090     A
GRA                ZR     X6,CUC2     IF ^(^D,PI,RJRS)[DI] & USES[DI] = 0        GRA        6091     A
GRA                                                                              GRA        6092     A
GRA                SB4    X3                                                     GRA        6093     A
GRA                SX7    B1                                                     GRA        6094     A
GRA                JP     B4          JUMP( ([UC.JT( OC[R1] )] )                 GRA        6095     A
GRA                                                                              GRA        6096     A
GRA                PROCESS DAR                                                   GRA        6097     A
GRA                MX3    -D.USESL                                               GRA        6098     A
GRA                BX6    -X3*X4                                                 GRA        6099     A
GRA                NZ     X6,TYI      IF USES[DI] " 0                            GRA        6100     A
GRA                                                                              GRA        6101     A
GRA      *         USELESS DEF, MARK AS DEAD                                     GRA        6102     A
GRA                                                                              GRA        6103     A
GRA       CUC2     SA6    A4          [DI] = 0                                   GRA        6104     A
GRA                SA0    A5          DD = R1                                    GRA        6105     A
GRA                EQ     CUC1                                                   GRA        6106     A
GRA                                                                              GRA        6107     A
GRA                PROCESS EOS                                                   GRA        6108     A
GRA                SX7    X5                                                     GRA        6109     A
GRA                ZR     X7,CUC1     IF IH[R1] = 0 */ NOT AN OPT=2 *EOS*        GRA        6110     A
GRA                                                                              GRA        6111     A
GRA                PROCESS NOP                                                   GRA        6112     A
GRA                MX6    0                                                      GRA        6113     A
GRA                EQ     CUC2                                                   GRA        6114     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                                                                                GRA        6115    A
GRA                 PROCESS DWL                                                    GRA        6116    A
GRA                 BX4    "RN"X5                                                  GRA        6117         I
          -CCGA039
GRA                 SA3    "TB"+X4     DIP = TB + RI[R1]                          GRA        6118         I
          -CCGA039
GRA                 MX6    0           PS = 0
>>>>      CCGA039 18                                                              CCGA039      11    A
GRA                 LX4    59-D.L2P
>>>>      CCGA039 19                                                              CCGA039      12    A
GRA                 MI     X4,CUC10    IF L2[DI]
>>>>      CCGA039 20                                                              CCGA039      13    A
GRA                 LX6    B6,X7       PS = 1
>>>>      CCGA039 21                                                              CCGA039      14    A
GRA       *
>>>>      CCGA039 22                                                              CCGA039      15    A
GRA        CUC10    BX4    "RN"X5
>>>>      CCGA039 23                                                              CCGA039      16    A
GRA                 SA3    "TB"+X4     DIP = TB + RI[R1]
>>>>      CCGA039 24                                                              CCGA039      17    A
GRA                 BX3    X3+X6       DIP = DIP + PS
>>>>      CCGA039 25                                                              CCGA039      18    A
GRA                 IX6    X3+X7       USES[DIP] = USES[DIP] + 1                  GRA        6119    A
GRA                 SA6    A3                                                     GRA        6120    A
GRA                 EQ     TYI                                                    GRA        6121    A
GRA                                                                               GRA        6122    A
GRA                 PROCESS (UP,NR,RNZ)                                            GRA        6123    A
GRA                 LX5    -R1.RKP                                                GRA        6124    A
GRA                 BX4    "RN"X5                                                 GRA        6125    A
GRA                 SA3    "TB"+X4     DK = TB + RK[R1]                           GRA        6126    A
GRA                 IX6    X3+X7       USES[DK] = USES[DK] + 1                    GRA        6127    A
GRA                 SA6    A3                                                     GRA        6128    A
GRA                 EQ     CUC1                                                   GRA        6129    A
GRA        RS       SPACE  3,14                                                   GRA        6130    A
GRA                 PROCESS RS                                                    GRA        6131    A
GRA                 LX5    -R1.SOP-SO.LKP                                         GRA        6132    A
GRA                 MX3    -SO.LKL                                                GRA        6133    A
GRA                 SA2    A5-B2       DIP = R1 - 2                               GRA        6134    A
GRA                 BX6    -X3*X5                                                 GRA        6135    A
GRA                 MX3    -D.USESL                                               GRA        6136    A
GRA                 SB3    X6          LKT = SOLK[R1]                             GRA        6137    A
GRA                 GT     B3,B2,RS2   IF LKT = 3    */ RJ RS                     GRA        6138    A
GRA                                                                               GRA        6139    A
GRA                 LX7    D.PRSP                                                 GRA        6140    A
GRA                 BX6    X7+X2       PRS[DIP] = 1                               GRA        6141    A
GRA                 LX7    D.PSP-D.PRSP                                           GRA        6142    A
GRA                 BX6    -X7*X6      PS[DIP] = 0                                GRA        6143    A
GRA                 EQ     B3,B1,RS1   IF LKT = 1    */ TEMP LOCK                 GRA        6144    A
GRA                                                                               GRA        6145    A
GRA                 SX7    B1                                                     GRA        6146    A
GRA                 IX6    X6+X7       USES[DIP] = USES[DIP] + 1                  GRA        6147    A
GRA                 SA6    A2                                                     GRA        6148    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                EQ     CUC1                                                GRA        6149     A
GRA                                                                           GRA        6150     A
GRA        RS1     BX7    -X3*X2                                              GRA        6151     A
GRA                SA6    A2                                                  GRA        6152     A
GRA                NZ     X7,CUC1     IF USES[DIP] " 0                        GRA        6153     A
GRA                                                                           GRA        6154     A
GRA                SA7    A4          [DI] = 0                                GRA        6155     A
GRA                SA7    A2          [DIP] = 0                               GRA        6156     A
GRA                SA5    A2-B2       R1 = DIP - 2                            GRA        6157     A
GRA                SA0    A2-B2       DD = R1                                 GRA        6158     A
GRA                EQ     CUC1                                                GRA        6159     A
GRA                                                                           GRA        6160     A
GRA        RS2     BX6    -X3*X4                                              GRA        6161     A
GRA                LX7    D.RJRSP                                             GRA        6162     A
GRA                ZR     X6,RS3      IF USES[DI] = 0                         GRA        6163     A
GRA                BX6    X7+X2       RJRS[DIP] = 1                           GRA        6164     A
GRA                SA6    A2                                                  GRA        6165     A
GRA                EQ     CUC1                                                GRA        6166     A
GRA                                                                           GRA        6167     A
GRA        RS3     SA6    A4          [DI] = 0                                GRA        6168     A
GRA                SA0    A5          DD = R1                                 GRA        6169     A
GRA                SA3    A2-B2                                               GRA        6170     A
GRA                BX7    -X7*X2      RJRS[DIP] = 0                           GRA        6171     A
GRA                LX0    R1.RJP                                              GRA        6172     A
GRA                SA7    A2                                                  GRA        6173     A
GRA                BX6    X0*X3       RJ[R1-4] = 0                            GRA        6174     A
GRA                LX0    -R1.RJP                                             GRA        6175     A
GRA                SA6    A3                                                  GRA        6176     A
GRA                EQ     CUC1                                                GRA        6177     A
GRA        DEF     SPACE  2                                                   GRA        6178     A
GRA                PROCESS DEF                                                GRA        6179     A
GRA                LX7    D.PSP                                               GRA        6180     A
GRA                BX6    -X7*X4      PS[DI] = 0                              GRA        6181     A
GRA                SA6    A4                                                  GRA        6182     A
GRA                EQ     CUC1                                                GRA        6183     A
GRA        BOS     SPACE  3,20                                                GRA        6184     A
GRA                PROCESS BOS                                                GRA        6185     A
GRA                MX1    -D.USESL                                            GRA        6186     A
GRA                BX6    X1*X4       USES[DI] = 0                            GRA        6187     A
GRA                SA6    A4                                                  GRA        6188     A
GRA                SB6    A0          DI = DD                                 GRA        6189     A
GRA                ZR     B6,CUC      IF SI = 0     */ NO USELESS DEFS        GRA        6190     A
GRA                                                                           GRA        6191     A
GRA      *         COMPRESS SEQUENCE TO REMOVE USELESS INSTRUCTIONS           GRA        6192     A
GRA                                                                           GRA        6193     A
GRA                SA4    A0+B2       DI = DD + 2                             GRA        6194     A
GRA                SB4    B2+B2                                               GRA        6195     A
GRA                                                                           GRA        6196     A
GRA        BOS1    SA4    A4+B4       DI = DI + 4                             GRA        6197     A
GRA                ZR     X4,BOS1     IF [DI] = 0                             GRA        6198     A
GRA                                                                           GRA        6199     A
GRA                SA5    A4-B2       R1 = DI - 2                             GRA        6200     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                 SA3     A4-B1       R2 = DI - 1                                 GRA         6201     A
GRA                 BX6     X4                                                      GRA         6202     A
GRA                 UX2     B3,X5                                                   GRA         6203     A
GRA                 SA6     B6+B2       [SI+2] = [DI]                               GRA         6204     A
GRA                 BX7     X5                                                      GRA         6205     A
GRA                 SA7     B6          [SI] = [R1]                                 GRA         6206     A
GRA                 BX6     X3                                                      GRA         6207     A
GRA                 SA6     B6+B1       [SI+1] = [R2]                               GRA         6208     A
GRA                 SB6     B6+B4       SI = SI + 4                                 GRA         6209     A
GRA                 NZ      B3,BOS1     IF OC[R1] " 0                               GRA         6210     A
GRA                                                                                 GRA         6211     A
GRA                 SB5     "TB"-B2                                                 GRA         6212     A
GRA                 SX7     B6-B5       L.TXT = SI - O.TXT                          GRA         6213     A
GRA                 SA7     L.TXT                                                   GRA         6214     A
GRA                 SX1     A5-B5       OTL = R1 - O.TXT  */ OLD LENGTH             GRA         6215     A
GRA                 ALLOC   RND,X1      ALLOC( RND , OTL )                          GRA         6216     A
GRA                 SA4     O.TXT                                                   GRA         6217     A
GRA                 SB7     X2                                                      GRA         6218     A
GRA                 MX2     0                                                       GRA         6219     A
GRA                 SA0     X4                                                      GRA         6220     A
GRA                 CALL    RNI         RENUMBER THE INSTRUCTIONS                   GRA         6221         I
          -CCGNC
GRA                 CALL    RNI#        RENUMBER THE INSTRUCTIONS                   CCGNC        668     A
GRA                 SX6     0                                                       GRA         6222     A
GRA                 SA6     L.RND       L.RND = 0                                   GRA         6223     A
GRA                 EQ      CUC                                                     GRA         6224     A
GRA       TYIII     SPACE   3,14                                                    GRA         6225     A
GRA                 PROCESS (ST,TST)                                                GRA         6226     A
GRA                 BX5     "RN"X5                                                  GRA         6227     A
GRA                 SA3     "TB"+X5     DIP = TB + RI[R1]                           GRA         6228     A
GRA                 LX6     B6,X7                                                   GRA         6229     A
GRA                 IX2     X3+X7       USES[DIP] = USES[DIP] + 1                   GRA         6230     A
GRA                 BX6     X6+X2       PS[DIP] = 1                                 GRA         6231     A
GRA                 SA6     A3                                                      GRA         6232     A
GRA                                                                                 GRA         6233     A
GRA                 PROCESS (LD,ILD,TLD,STT,KLS,KRS)                                GRA         6234         I
          -CCG1
GRA                 PROCESS (LD,ILD,TLD,LDC,STT,KLS,KRS)                            CCG1         151         I
          -CCG16
GRA                 PROCESS (LD,ILD,TLD,LDC,LDV,STT,KLS,KRS)                        CCG16        241     A
GRA                 SA3     A5+B1       R2 = R1 + 1                                 GRA         6235     A
GRA                 LX3     -IH.RFP                                                 GRA         6236     A
GRA                 BX6     "RN"X3                                                  GRA         6237     A
GRA                 SA2     "TB"+X6     DJ = TB + RF[R2]                            GRA         6238     A
GRA                 IX6     X2+X7       USES[DJ] = USES[DJ] + 1                     GRA         6239     A
GRA                 SA6     A2                                                      GRA         6240     A
GRA                 EQ      CUC1                                                    GRA         6241     A
GRA       JPX       SPACE   2,14                                                    GRA         6242     A
GRA                 PROCESS JPBB                                                    GRA         6243         I
          -CCG2
GRA                 PROCESS (JPBB,RJXJ)                                             CCG2          19     A
GRA                 SA3     A5+B1       R2 = R1 + 1                                 GRA         6244     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

```
GRA                LX3    -IH.RFP                                          GRA        6245     A
GRA                BX6    "RN"X3                                           GRA        6246     A
GRA                SA2    "TB"+X6     DJ = TB + RF[R2]                     GRA        6247     A
GRA                IX6    X2+X7       USES[DJ] = USES[DJ] + 1              GRA        6248     A
GRA                SA6    A2                                               GRA        6249     A
GRA                                                                       GRA        6250     A
GRA                PROCESS (JIN,JPX)                                       GRA        6251     A
GRA                BX5    "RN"X5                                           GRA        6252     A
GRA                SA3    "TB"+X5     DIP = TB + RI[R1]                    GRA        6253     A
GRA                IX6    X3+X7       USES[DIP] = USES[DIP] + 1            GRA        6254     A
GRA                SA6    A3                                               GRA        6255     A
GRA                EQ     CUC1                                             GRA        6256     A
GRA                                                                       GRA        6257     A
GRA                QUAL                                                    GRA        6258     A
GRA        IRA     EQU    /IRA/IRA                                         GRA        6259     A
GRA        CUC     EQU    /CUC/CUC                                         GRA        6260     A
GRA        CUC#    EQENT  CUC                                              CCGNC       669     A
GRA        OPR     SPACE  2,14                                             GRA        6261     A
GRA       **       OPR - DEFINE MACRO TO CREATE COMBINED JUMP TABLE FOR *IRA/CUC*  GRA  6262  A
GRA                                                                       GRA        6263     A
GRA                MACRO  OPR,NAM,SS                                       GRA        6264     A
GRA                NOREF  /IRA/.NAM,/CUC/.NAM                              GRA        6265     A
GRA        TYP     MICRO  4,1,$SS$                                         GRA        6266     A
GRA        O       IF     DEF,/IRA/.NAM                                    GRA        6267     A
GRA                EQ     /IRA/.NAM           NAM                          GRA        6268     A
GRA        O       ELSE                                                    GRA        6269     A
GRA                IFC    EQ,/"TYP"/ /,2                                   GRA        6270     A
GRA                EQ     /IRA/TYI    NAM                                  GRA        6271     A
GRA                SKIP   1                                                GRA        6272     A
GRA                EQ     ERROR                                            GRA        6273     A
GRA        O       ENDIF                                                   GRA        6274     A
GRA       *                                                               GRA        6275     A
GRA        O       IF     DEF,/CUC/.NAM                                    GRA        6276     A
GRA       -        JP     /CUC/.NAM           NAM                          GRA        6277     A
GRA        O       ELSE                                                    GRA        6278     A
GRA                IFC    EQ,/"TYP"/ /,2                                   GRA        6279     A
GRA       -        JP     /CUC/TYI    NAM                                  GRA        6280     A
GRA                SKIP   1                                                GRA        6281     A
GRA       -        JP     377777B                                          GRA        6282     A
GRA        O       ENDIF                                                   GRA        6283     A
GRA                ENDM                                                    GRA        6284     A
GRA                                                                       GRA        6285     A
GRA        IRA.JT  BSS    0                                                GRA        6286     A
GRA        CUC.JT  BSS    0                                                GRA        6287     A
GRA       *CALL    OPRDEFS                                                 GRA        6288     A
GRA                                                                       GRA        6289     A
GRA                END                                                     GRA        6290     A
```

SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - GRA

```
IDENTIFIER                TOTAL    ACTIVE

0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

GRA	6290	5629
FCC0392	9	8
FCC0414	77	74
FCC0448	74	38
FCC0513	31	28
FEAT121A	2	0
FCC0514	117	116
FCC0535	9	9
FCC0594	20	15
FCC0603	80	76
FCC0645	24	23
FCC0646	56	55
FCC0651	12	12
FCC0656	3	3
FCC0675	66	65
FCC0735	22	22
FCC0764	15	15
FCC0811	40	40
FCC0812	10	10
FCC0814	3	3
FCC0839	3	3
FCC0882	4	4
FCC0912	26	24
FCC0919	4	4
FCC0777	6	0
FCC0909	50	48
FCC0914	88	88
FCC0977	31	30
FCC1014	100	97
FCC0985	51	41
FCC1039	28	28
FCC1040	6	6
FCC1107	7	7
FCC1127	24	24
FCC1196	6	6
FCC1227	9	8
FCC1491X	4	2
FCC1531X	3	3
FCC1518X	12	10
FCC1568X	3	2
CCG1	63	43
CCG2	7	7
CCG4	14	4
FCC1626	5	5
FCC1644X	16	16
FCC1646X	21	21
CCG7	3	2
CCGNC	28	28
CCG11	2	2
FCC1740X	12	11
FCC1770X	12	12
FCC1828X	4	4

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN GRA

CCG16	37	36
CCG18	12	12
FCC1918	3	3
CCG21	15	1
CCG23	17	17
CCG24	22	22
FCC1997	11	11
CCG25	181	178
CCG28	22	17
FCC2056	10	10
FCC2051	14	13
CCG34	17	16
FCC2074	8	8
FCC2092	3	3
FCC2124	79	76
FCC2135	63	63
FCC2165	1	1
CCG41	10	7
CCGA009	6	6
FCCA509X	17	17
CCGA036	1	1
CCGA039	14	14
CCGA062	6	1
CCGA063	3	3
CCG0018	7	7
CCG0022	2	2
CCGA082	10	10
CCG0037	3	3
CCG0045	6	6
CCG0071	22	22
CCG0012	5	5
CCG0034	23	23
CCG0057	9	9
CCG0089	10	10
CCG0093	2	2
CCG0117	2	2
CCG0119	15	15
CCG0128	13	13
CCG0146	1	1
CCG0172	12	12
CCGA097	46	46
CCGA099	15	1
CCGA100	13	13

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ   *DECK     PROSEQ                                                    PROSEQ         1     A
PROSEQ             IDENT  PROSEQ                                             PROSEQ         2     A
PROSEQ    PROSEQ   TITLE  PROSEQ - PROCESS ACCUMULATED SEQUENCES              PROSEQ         3     A
PROSEQ   *CALL     SSTCALL                                                   PROSEQ         4     A
PROSEQ    B=PROSQ  RPVDEF                                                    PROSEQ         5     A
PROSEQ             PASS2TM                                                   PROSEQ         6           I

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

          -CCG1
PROSEQ                                                                    PROSEQ        7          I
          -CCG1
PROSEQ              EXT    BRN,WB                                         PROSEQ        8          I
          -CCG1
PROSEQ                                                                    PROSEQ        9     A
PROSEQ    **        PROSEQ - PROCESS ACCUMULATED SEQUENCES                PROSEQ       10     A
PROSEQ     FLAGS    SPACE  3,14                                           PROSEQ       11     A
PROSEQ     BIR      ENTRY. 100002B     BASE INTERMEDIATE R-NUMBER         PROSEQ       12          I
          -CCG1
PROSEQ     NIRN     ENTRY. 100002B     NEXT INTERMEDIATE R-NUMBER         PROSEQ       13          I
          -CCG1
PROSEQ     .ROL     IFNE   HC.ROL,0,2                                     CCG1        152     A
PROSEQ     CG.BIR   ENTRY. 100002B     BASE INTERMEDIATE R-NUMBER         CCG1        153          I
          -CCGNC
PROSEQ     CG.NIRN  ENTRY. 100002B     NEXT INTERMEDIATE R-NUMBER         CCG1        154          I
          -CCGNC
PROSEQ                                                                    PROSEQ       14          I
          -CCGNC
PROSEQ              ENTRY  CG.BRN                                         CCG5        879          I
          -CCGNC
PROSEQ     CG.BRN   DATA   4,4                                            CCG5        880          I
          -CCGNC
PROSEQ                                                                    CCG5        881          I
          -CCGNC
PROSEQ     CG.LNL   ENTRY. 0           "0 IF WE ARE PROCESSING PART OF A LOOP  CCG5  882          I
          -CCGNC
PROSEQ     CG.OPTL  ENTRY. 0           "0 IF *TXT* CONTAINS THE BODY OF AN     CCG5  883          I
          -CCGNC
PROSEQ    *                            OPTIMIZABLE LOOP                      CCG5        884          I
          -CCGNC
PROSEQ     CG.PIOL  ENTRY. 0           PREVIOUS LENGTH OF *IOL*              CCG5        885          I
          -CCGNC
PROSEQ     FPL2     ENTRY. 0           "0 IF F.P.S OR LEVEL 2 SYMBOLS IN SUBPROGRAM  PROSEQ  15     I
          -CCG1     -CCGNC
PROSEQ     LALS     BSSZ   1           L.ALS-1 OF LAST SEQUENCE              PROSEQ        16          I
          -CCG1     -CCGNC
PROSEQ     PROSEQ   SPACE  3           PAS                                   PROSEQ        17          I
          -CCG1     -CCGNC
PROSEQ     PROSEQ   ENTRY.                                                   PROSEQ        18          I
          -CCG1     -CCGNC
PROSEQ     CG.SRF   ENTRY. 0           "0 IF FP OR LEVEL 2 SYMBOLS IN PROCEDURE  CCG1      155          I
          -CCGNC
PROSEQ     CG.XR    ENTRY. 0           =1 IF EXT REFS IN CURRENT SEQ         CCG7        116          I
          -CCGNC
PROSEQ                                                                    CCG7        117          I
          -CCGNC
PROSEQ              ECHO   1,Z=(BLK,TXT,MLT,MOD,SEQ)                      CCG5        886          I
          -CCGNC
PROSEQ              EXT    O.Z,L.Z                                        CCG5        887          I
          -CCGNC
PROSEQ     CC$BIR   ENTRY. 100002B     BASE INTERMEDIATE R-NUMBER         CCGNC       670     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ    CC$NIRN  ENTRY. 100002B     NEXT INTERMEDIATE R-NUMBER                    CCGNC        671     A
PROSEQ                                                                              CCGNC        672     A
PROSEQ             ENTRY  CC$BRN                                                    CCGNC        673     A
PROSEQ    CC$BRN   DATA   4,4                                                       CCGNC        674     A
PROSEQ                                                                              CCGNC        675     A
PROSEQ    CC$LNL   ENTRY. 0           "0 IF PROCESSING PART OF A LOOP               CCGNC        676     A
PROSEQ    CC$OPF   ENTRY. 1           OPT=2 PHASE FLAG, =0 IF OPT=2 AND BRIDGE      CCG27          6     A
PROSEQ    CC$OPTL  ENTRY. 0           "0 IF *TXT* CONTAINS THE BODY OF AN           CCGNC        677     A
PROSEQ   *                            OPTIMIZABLE LOOP                              CCGNC        678     A
PROSEQ    CC$PIOL  ENTRY. 0           PREVIOUS LENGTH OF *IOL*                      CCGNC        679     A
PROSEQ    CC$SRF   ENTRY. 0           "0 IF FP OR LEVEL 2 SYMBOLS IN PROCEDURE      CCGNC        680     A
PROSEQ    CC$XR    ENTRY. 0           =1 IF EXT REFS IN CURRENT SEQ                 CCGNC        681     A
PROSEQ                                                                              CCGNC        682     A
PROSEQ             TABLES (BLK,MLT,MOD,RCT,RND,PIT,SEQ,TXT)                         CCGNC        683     A
PROSEQ             EXT#   (LM,TYIOM)                                                CCGNC        684     A
PROSEQ    CG.CPC   SPACE  3,14                                                      CCG5         888         I
         -CCGNC
PROSEQ    CG$CPC   SPACE  3,14                                                      CCGNC        685     A
PROSEQ   **        CG.CPC - CODE PROLOGUE CODE                                     CCG5         889         I
         -CCGNC
PROSEQ   **        CG$CPC - CODE PROLOGUE CODE                                     CCGNC        686     A
PROSEQ   *                                                                          CCG5         890     A
PROSEQ   *         ENTRY  SAME CALLING SEQUENCE AS CG.PAS.                         CCG5         891         I
         -CCGNC
PROSEQ   *         ENTRY  SAME CALLING SEQUENCE AS CG$PAS.                         CCGNC        687     A
PROSEQ   *                                                                          CCG5         892     A
PROSEQ   *         EXIT   SEQUENCE CODED AND WRITTEN TO *SLIST* FILE.              CCG5         893     A
PROSEQ                                                                              CCG5         894     A
PROSEQ    CG.CPC   ENTRY.                                                           CCG5         895         I
         -CCGNC
PROSEQ    CG$CPC   ENTRY.                                                           CCGNC        688     A
PROSEQ             SA1    =XCO.OPT                                                 CCG5         896         I
         -CCGNC
PROSEQ             SA1    =XHO$OPT                                                 CCGNC        689     A
PROSEQ             LX1    2                                                        CCG5         897     A
PROSEQ             BX6    X1          CO.OPT = SHIFT(CO.OPT,2)  */ FORCE OPT"2     CCG5         898         I
         -CCGNC
PROSEQ             BX6    X1          HO$OPT = SHIFT(HO$OPT,2)  */ FORCE OPT"2     CCGNC        690     A
PROSEQ             SA6    A1                                                       CCG5         899     A
PROSEQ             CALL   CG.PAS                                                   CCG5         900         I
         -CCGNC
PROSEQ             CALL   CG$PAS                                                   CCGNC        691     A
PROSEQ             SA1    CO.OPT                                                   CCG5         901         I
         -CCGNC
PROSEQ             SA1    HO$OPT                                                   CCGNC        692     A
PROSEQ             LX1    -2                                                       CCG5         902     A
PROSEQ             BX6    X1          CO.OPT = SHIFT(CO.OPT,-2)  */ RESTORE        CCG5         903         I
         -CCGNC
PROSEQ             BX6    X1          HO$OPT = SHIFT(HO$OPT,-2)  */ RESTORE        CCGNC        693     A
PROSEQ             SA6    A1                                                       CCG5         904     A
PROSEQ             EQ     CG.CPC                                                   CCG5         905         I
         -CCGNC

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            EQ     CG$CPC                                                  CCGNC        694     A
PROSEQ    CG.PAS  SPACE  3           PAS                                         CCG1         156         I
         -CCGNC
PROSEQ    CG$PAS  SPACE  3           PAS                                         CCGNC        695     A
PROSEQ   **       CG.PAS - PROCESS ACCUMULATED SEQUENCES                         CCG1         157         I
         -CCGNC
PROSEQ   **       CG$PAS - PROCESS ACCUMULATED SEQUENCES                         CCGNC        696     A
PROSEQ                                                                           CCG1         158     A
PROSEQ    CG.PAS  ENTRY.                                                         CCG1         159         I
         -CCGNC
PROSEQ    CG$PAS  ENTRY.                                                         CCGNC        697     A
PROSEQ            MX7    0                                                       PROSEQ        19     A
PROSEQ            SA7    =XL.RND     L.RND = 0                                   PROSEQ        20         I
         -CCGNC
PROSEQ            SA7    L.RND       L.RND = 0                                   CCGNC        698     A
PROSEQ            ALLOC  TXT,4                                                   PROSEQ        21     A
PROSEQ            SA5    F.RDT+OC.EOQ                                            PROSEQ        22     A
PROSEQ            SX6    1S10+OC.EOQ TERMINATE *TXT* WITH A *EOQ*                PROSEQ        23     A
PROSEQ            LX6    R1.OCP                                                  PROSEQ        24     A
PROSEQ            SA6    X2+B6                                                   PROSEQ        25     A
PROSEQ            MX7    0           [R2] = 0                                    PROSEQ        26     A
PROSEQ            SA7    A6+B1                                                   PROSEQ        27     A
PROSEQ            BX6    X5                                                      PROSEQ        28     A
PROSEQ            SA1    BRN+1                                                   PROSEQ        29         I
         -CCG1
PROSEQ            SA1    =XCG.BRN+1                                              CCG1         160         I
         -CCG5
PROSEQ            SA6    A7+B1                                                   PROSEQ        30     A
PROSEQ            SA7    A6+B1                                                   PROSEQ        31     A
PROSEQ    .ROL    IFNE   HC.ROL,0                                                CCG5         906     A
PROSEQ            SA1    =XCG.BRN+1                                              CCG5         907         I
         -CCGNC
PROSEQ            SA1    =XCC$BRN+1                                              CCGNC        699     A
PROSEQ            MX0    -R1.RIL                                                 PROSEQ        32     A
PROSEQ            BX6    -X0*X1      SET MAX R-NUMBER FOR *SQZ*                  PROSEQ        33     A
PROSEQ            SA6    A1                                                      PROSEQ        34     A
PROSEQ    .ROL    ENDIF                                                          CCG5         908     A
PROSEQ                                                                           PROSEQ        35     A
PROSEQ    .AET    IF     DEF,WB.AETL                                             CCG5         909     A
PROSEQ            SA3    FPL2                                                    PROSEQ        36         I
         -CCG1
PROSEQ            ZR     X3,PAS1     IF N.FP & LEVEL2 = 0                        PROSEQ        37         I
         -CCG1
PROSEQ            SA3    CG.SRF                                                  CCG1         161         I
         -CCGNC
PROSEQ            SA3    CC$SRF                                                  CCGNC        700     A
PROSEQ            ZR     X3,PAS1     IF N.FPS & LEVEL2 = 0                       CCG1         162         I
         -CCG5
PROSEQ            RJ     PSB         PROPOGATE SPECIAL BITSS TO DESCRIPTOR       PROSEQ        38         I
         -CCG5
PROSEQ                                                                           PROSEQ        39         I
         -CCG5

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ    PAS1     CALL   SBB         ELIMINATE REDUNDANT OPERATIONS                    PROSEQ        40          I
          -CCG5
PROSEQ    +        ZR     X3,*+1      IF CG.SRF = 0 */ NO SPECIAL REFS                  CCG5         910          I
          -CCGNC
PROSEQ    +        ZR     X3,*+1      IF CC$SRF = 0 */ NO SPECIAL REFS                  CCGNC        701     A
PROSEQ                                                                                   CCG5         911     A
PROSEQ             RJ     PSB         PROPAGATE BITS IN *AET* TO IL INSTRUCTIONS        CCG5         912     A
PROSEQ    .AET     ENDIF                                                                CCG5         913     A
PROSEQ                                                                                   CCG5         914     A
PROSEQ             CALL   SBB         ELIMINATE REDUNDANT INSTRUCTIONS                  CCG5         915          I
          -CCGNC
PROSEQ             CALL   SBB#        ELIMINATE REDUNDANT INSTRUCTIONS                  CCGNC        702     A
PROSEQ             IFNE   HC.STP,0,1                                                    CCG16        242     A
PROSEQ             CALL   BR$AFT      ADJUST FUNCTION TEMPS                             CCG16        243     A
PROSEQ             SA1    =XOPT2                                                        PROSEQ        41          I
          -CCG1
PROSEQ             SA1    =XCO.OPT                                                      CCG1         163          I
          -CCGNC
PROSEQ             SA1    =XHO$OPT                                                      CCGNC        703     A
PROSEQ             SA5    L.TXT                                                         PROSEQ        42     A
PROSEQ             NZ     X1,PAS5     IF OPT2 " 0                                       PROSEQ        43          I
          -CCG1
PROSEQ             MI     X1,PAS5     IF OPT = 2                                        CCG1         164     A
PROSEQ                                                                                   PROSEQ        44     A
PROSEQ             SX7    X5-8                                                          PROSEQ        45     A
PROSEQ             ZR     X7,PAS6     IF L.TXT = 8  */ EMPTY SEQUENCE                   PROSEQ        46     A
PROSEQ             CALL   FSU                                                           FEAT155A      81          I
          -FEAT155AB
PROSEQ             SA1    WB                                                            PROSEQ        47          I
          -FEAT155AB
PROSEQ             SA1    =XCO.ER                                                       FEAT155AB      2          I
          -CCGNC
PROSEQ             SA1    =XHO$ER                                                       CCGNC        704     A
PROSEQ             ZR     X1,PAS0     IF ER = 0      */ NO OTR                          FEAT155AB      3     A
PROSEQ             CALL   FSU         FORCE UPPER                                       FEAT155AB      4          I
          -CCGNC
PROSEQ             CALL   FSU#        FORCE UPPER                                       CCGNC        705     A
PROSEQ    PAS0     SA1    WB                                                            FEAT155AB      5          I
          -CCG1
PROSEQ             SX6    X1-1                                                          PROSEQ        48          I
          -CCG1
PROSEQ             NZ     X6,PAS1A    IF WB " 1                                         PROSEQ        49          I
          -CCG1
PROSEQ                                                                                   PROSEQ        50          I
          -CCG1
PROSEQ             SA2    LM                                                            PROSEQ        51          I
          -CCG1
PROSEQ             SA3    O.TEND                                                        PROSEQ        52          I
          -CCG1
PROSEQ             IX4    X3-X2       WS = O.TEND - LM                                  PROSEQ        53          I
          -CCG1
PROSEQ             AX4    3                                                             PROSEQ        54          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
          -CCG1
PROSEQ              AX3    B1,X4                                          PROSEQ        55          I
          -CCG1
PROSEQ              IX4    X3+X4                                          PROSEQ        56          I
          -CCG1
PROSEQ              IX7    X5-X4                                          PROSEQ        57          I
          -CCG1
PROSEQ              PL     X7,PAS1A    IF L.TXT > WS * 3/16               PROSEQ        58          I
          -CCG1
PROSEQ     PAS0     SA1    =XCG.OPTL                                      CCG1         165          I
          -CCGNC
PROSEQ     PAS0     SA1    =XCC$OPTL                                      CCGNC        706     A
PROSEQ              ZR     X1,PAS1A    IF CG.OPTL = 0  */ NOT AN OPTIMIZABLE LOOP  CCG1  166          I
          -CCGNC
PROSEQ              ZR     X1,PAS1A    IF CG$OPTL = 0  */ NOT AN OPTIMIZABLE LOOP  CCGNC  707     A
PROSEQ                                                                    PROSEQ        59     A
PROSEQ              CALL   COL         CODE OPTIMIZABLE LOOP ( OPT=1 )    PROSEQ        60          I
          -CCGNC
PROSEQ              CALL   COL#        CODE OPTIMIZABLE LOOP ( OPT=1 )    CCGNC        708     A
PROSEQ              EQ     PAS6                                           PROSEQ        61     A
PROSEQ                                                                    PROSEQ        62     A
PROSEQ     PAS1A    SA1    FPL2                                           PROSEQ        63          I
          -CCG1
PROSEQ     PAS1A    SA1    CG.SRF                                         CCG1         167          I
          -CCG5
PROSEQ              ZR     X1,PAS4     IF FPL2 = 0   */ NO F.P.S OF LEVEL IN PROG  PROSEQ  64          I
          -CCG1     -CCG5
PROSEQ              ZR     X1,PAS4     IF CG.SRF = 0   */ NO F.P.S OF LEVEL IN PRO  CCG1  168          I
          -CCG5
PROSEQ     PAS1A    BSS    0                                              CCG5         916     A
PROSEQ     .AET     IF     DEF,WB.AETL                                    CCG5         917     A
PROSEQ              SA1    CG.SRF                                         CCG5         918          I
          -CCGNC
PROSEQ              SA1    CC$SRF                                         CCGNC        709     A
PROSEQ              ZR     X1,PAS4     IF CG.SRF = 0 */ NO SPECIAL REFS IN PROC  CCG5  919          I
          -CCGNC
PROSEQ              ZR     X1,PAS4     IF CC$SRF = 0 */ NO SPECIAL REFS IN PROC  CCGNC  710     A
PROSEQ                                                                    PROSEQ        65     A
PROSEQ              RJ     SSP         SET *SEQ* POINTERS                 PROSEQ        66     A
PROSEQ              SA1    LALS                                           PROSEQ        67          I
          -CCG5
PROSEQ              SA1    CG.LNL                                         CCG5         920          I
          -CCGNC
PROSEQ              SA1    CC$LNL                                         CCGNC        711     A
PROSEQ              SX5    2                LCM                           PROSEQ        68     A
PROSEQ              NZ     X1,PAS2     IF LALS " 0   */ IN A LOOP         PROSEQ        69          I
          -CCG5
PROSEQ              NZ     X1,PAS2     IF CG.LNL " 0 */ IN A LOOP         CCG5         921          I
          -CCGNC
PROSEQ              NZ     X1,PAS2     IF CG$LNL " 0 */ IN A LOOP         CCGNC        712     A
PROSEQ              SX5    3                                              PROSEQ        70     A
PROSEQ     PAS2     MX6    1                                              PROSEQ        71     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            CALL   ESR                                                  PROSEQ        72     A
PROSEQ            SA1    L.MLT                                                PROSEQ        73     A
PROSEQ            ZR     X1,PAS3     IF L.MLT = 0  */ NO MODS TO BLOCK        PROSEQ        74     A
PROSEQ            SX6    1                                                    PROSEQ        75     A
PROSEQ            CALL   MPB         MERGE BLOCK AND MODS                     PROSEQ        76           I
          -CCGNC
PROSEQ            CALL   MPB#        MERGE BLOCK AND MODS                     CCGNC        713     A
PROSEQ            EQ     PAS4                                                 PROSEQ        77     A
PROSEQ                                                                        PROSEQ        78     A
PROSEQ    PAS3    SA1    O.SEQ                                                PROSEQ        79     A
PROSEQ            SA2    L.SEQ                                                PROSEQ        80     A
PROSEQ            BX6    X1                                                   PROSEQ        81     A
PROSEQ            SA6    O.TXT       O.TXT = O.SEQ                            PROSEQ        82     A
PROSEQ            LX7    X2                                                   PROSEQ        83     A
PROSEQ            SA7    L.TXT       L.TXT = L.SEQ                            PROSEQ        84     A
PROSEQ            MX6    0                                                    PROSEQ        85     A
PROSEQ            SA6    L.SEQ                                                PROSEQ        86     A
PROSEQ    .AET    ENDIF                                                       CCG5         922     A
PROSEQ                                                                        PROSEQ        87     A
PROSEQ    PAS4    SX6    0                                                    PROSEQ        88     A
PROSEQ            SA6    L.BLK                                                PROSEQ        89     A
PROSEQ            CALL   MCG         CODE THE SEQUENCE                        PROSEQ        90           I
          -CCGNC
PROSEQ            CALL   CII         CONVERT THE ISSUED INSTRUCTIONS          PROSEQ        91           I
          -CCG1   -CCGNC
PROSEQ            CALL   WII         CONVERT THE ISSUED INSTRUCTIONS          CCG1         169           I
          -CCGNC
PROSEQ            CALL   MCG#        CODE THE SEQUENCE                        CCGNC        714     A
PROSEQ            CALL   WII#        SAVE THE ISSUED INSTRUCTIONS             CCGNC        715     A
PROSEQ            EQ     PAS6                                                 PROSEQ        92     A
PROSEQ                                                                        PROSEQ        93     A
PROSEQ    PAS5    CALL   PBB         PROCESS BASIC BLOCK FOR OPT=2            PROSEQ        94           I
          -CCGNC
PROSEQ    PAS5    CALL   PBB#        PROCESS BASIC BLOCK FOR OPT=2            CCGNC        716     A
PROSEQ    PROSEQ  SPACE  3,8                                                  PROSEQ        95           I
          -CCG1
PROSEQ    CG.PAS  SPACE  3,8                                                  CCG1         170           I
          -CCGNC
PROSEQ    CG$PAS  SPACE  3,8                                                  CCGNC        717     A
PROSEQ    *       RESET FLAGS FOR NEXT SEQUENCE                               PROSEQ        96     A
PROSEQ                                                                        PROSEQ        97     A
PROSEQ    PAS6    SA5    LM                                                   PROSEQ        98           I
          -CCG5
PROSEQ    PAS6    SA5    =XLM                                                 CCG5         923     A
PROSEQ            MX6    0                                                    PROSEQ        99     A
PROSEQ            BX7    X5                                                   PROSEQ       100     A
PROSEQ            SA6    =XXR        XR = 0                                   PROSEQ       101           I
          -CCG1
PROSEQ            SA6    L.BLK       L.LBK = 0                                PROSEQ       102           I
          -CCG1
PROSEQ            SA6    L.BLK       L.BLK = 0                                CCG1         171     A
PROSEQ            SA6    CG.XR       CG.XR = 0                                CCG7         118           I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

          -CCGNC
PROSEQ            SA6    =XCC$XR     CC$XR = 0                                          CCGNC        718     A
PROSEQ            SA6    CG.PIOL     CG.PIOL = 0                                        CCG7         119          I
          -CCGNC
PROSEQ            SA6    CC$PIOL     CC$PIOL = 0                                        CCGNC        719     A
PROSEQ            SA6    CC$OPTL     CC$OPTL = 0                                        CCG12          8     A
PROSEQ            SA7    O.BLK       O.BLK = LM                                         PROSEQ       103     A
PROSEQ            SA7    O.TXT       O.TXT = LM                                         PROSEQ       104     A
PROSEQ            SX6    4                                                              PROSEQ       105     A
PROSEQ            SA6    L.TXT       L.TXT = 4                                          PROSEQ       106     A
PROSEQ            SA4    =XLSN       SETUP A *BOS* AT THE BEGIN OF *TXT*                PROSEQ       107          I
          -CCG5
PROSEQ            SA4    =XCO.CSN                                                       CCG5         924          I
          -CCGNC
PROSEQ            SA4    =XHO$CSN                                                       CCGNC        720     A
PROSEQ            LX4    R1.INP                                                         PROSEQ       108     A
PROSEQ            PX6    B1,X4                                                          PROSEQ       109     A
PROSEQ            SA4    F.RDT+B1                                                       PROSEQ       110     A
PROSEQ            SA6    X5                                                             PROSEQ       111     A
PROSEQ            MX7    0                                                              PROSEQ       112     A
PROSEQ            SA7    A6+B1                                                          PROSEQ       113     A
PROSEQ            BX6    X4                                                             PROSEQ       114     A
PROSEQ            SA6    A7+B1                                                          PROSEQ       115     A
PROSEQ            SA7    A6+B1                                                          PROSEQ       116          I
          -CCG1
PROSEQ     .ROL   IFNE   HC.ROL,0                                                       CCG1         172     A
PROSEQ            SA1    CG.BIR                                                         CCG1         173          I
          -CCGNC
PROSEQ            SA1    CC$BIR                                                         CCGNC        721     A
PROSEQ            SA1    BIR                                                            PROSEQ       117          I
          -CCG1
PROSEQ            SA2    BRN+1                                                          PROSEQ       118          I
          -CCG1
PROSEQ            SA2    CG.BRN+1                                                       CCG1         174          I
          -CCGNC
PROSEQ            SA2    CC$BRN+1                                                       CCGNC        722     A
PROSEQ            SA3    WB                                                             PROSEQ       119          I
          -CCG1
PROSEQ            BX6    X1                                                             PROSEQ       120     A
PROSEQ            SA6    A1+B1       BIR(2) = BIR(1)  ( INTERMEDIATE R-NUMBERS )        PROSEQ       121          I
          -CCG1
PROSEQ            SA6    A1+B1       CG.BIR(2) = CG.BIR(1)  ( INTERMEDIATE R-NUM        CCG1         175          I
          -CCGNC
PROSEQ            SA6    A1+B1       CC$BIR(2) = CC$BIR(1)  ( INTERMEDIATE R-NUM        CCGNC        723     A
PROSEQ            LX7    X2                                                             PROSEQ       122     A
PROSEQ            SA7    A2-B1       BRN(1) = BRN(2) ( NEW BASE = OLD MAX )             PROSEQ       123          I
          -CCG1
PROSEQ            SA7    A2-B1       CG.BRN(1) = CG.BRN(2) ( NEW BASE = OLD MAX         CCG1         176          I
          -CCGNC
PROSEQ            SA7    A2-B1       CC$BRN(1) = CC$BRN(2) ( NEW BASE = OLD MAX         CCGNC        724     A
PROSEQ            AX4    B1,X3                                                          PROSEQ       124          I
          -CCG1

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            BX6    X3*X4       WB = WB & SHIFT(WB,-1)  */ WB = 1S59 IF OPT=0     PROSEQ     125        I
          -CCG1
PROSEQ            SA5    L.ALS                                                         PROSEQ     126        I
          -CCG1
PROSEQ            SX7    X5-1                                                          PROSEQ     127        I
          -CCG1
PROSEQ            SA6    A3                                                            PROSEQ     128        I
          -CCG1
PROSEQ            SA7    LALS        LALS = L.ALS - 1                                  PROSEQ     129        I
          -CCG1
PROSEQ            EQ     PROSEQ                                                        PROSEQ     130        I
          -CCG1
PROSEQ    .ROL    ELSE                                                                 CCG1       177   A
PROSEQ            SX6    4                                                             CCG1       178   A
PROSEQ            SA6    =XCG.BRN+1  CG.BRN(2) = 4                                     CCG1       179        I
          -CCGNC
PROSEQ            SA6    =XCC$BRN+1  CC$BRN(2) = 4                                     CCGNC      725   A
PROSEQ    .ROL    ENDIF                                                                CCG1       180   A
PROSEQ            EQ     CG.PAS                                                        CCG1       181        I
          -CCGNC
PROSEQ            EQ     CG$PAS                                                        CCGNC      726   A
PROSEQ    SSP     TITLE  SSP - SET SEQUENCE POINTERS FOR OPT=1                         PROSEQ     131   A
PROSEQ    .AET    IF     DEF,WB.AETL                                                   CCG5       925   A
PROSEQ                                                                                 CCG5       926   A
PROSEQ    **      SSP - SET SEQUENCE POINTERS FOR OPT=1                                PROSEQ     132   A
PROSEQ    *                                                                            PROSEQ     133   A
PROSEQ    *       ENTRY  SEQUENCE IN *TXT*                                             PROSEQ     134   A
PROSEQ    *                                                                            PROSEQ     135   A
PROSEQ    *       EXIT   SEQUENCE COPIED TO *BLK*                                      PROSEQ     136   A
PROSEQ    *              (O.SEQ,L.SEQ) = FWA,LENGTH OF SEQUENCE                        PROSEQ     137   A
PROSEQ                                                                                 PROSEQ     138   A
PROSEQ            EXT    O.SEQ,L.SEQ                                                   PROSEQ     139        I
          -CCGNC
PROSEQ                                                                                 PROSEQ     140   A
PROSEQ    SSP     ENTRY. **                                                            PROSEQ     141        I
          -CCGNC
PROSEQ    SSP     ENTRY. **,#                                                          CCGNC      727   A
PROSEQ            SA4    L.BLK                                                         PROSEQ     142   A
PROSEQ            NZ     X4,SSP1     IF L.BLK " 0                                      PROSEQ     143   A
PROSEQ                                                                                 PROSEQ     144   A
PROSEQ            SA5    LM                                                            PROSEQ     145   A
PROSEQ            ALLOC  TXT,4       ALLOC( TXT , 4 )                                  PROSEQ     146   A
PROSEQ            BX6    X2                                                            PROSEQ     147   A
PROSEQ            SA6    O.SEQ       O.SEQ = O.TXT                                     PROSEQ     148   A
PROSEQ            SX7    X2+B6                                                         PROSEQ     149   A
PROSEQ            SA7    A2          O.TXT = O.TXT + L.TXT-4                           PROSEQ     150   A
PROSEQ            SX6    B6                                                            PROSEQ     151   A
PROSEQ            SA6    L.SEQ       L.SEQ = L.TXT - 4                                 PROSEQ     152   A
PROSEQ            SA6    X2+B1       TXTL[SEQ+1] = L.SEQ                               PROSEQ     153   A
PROSEQ            SX6    4                                                             PROSEQ     154   A
PROSEQ            SA6    A3          L.TXT = 4                                         PROSEQ     155   A
PROSEQ            IX7    X7-X5                                                         PROSEQ     156   A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            SA7    L.BLK       L.BLK = O.TXT - LM                    PROSEQ     157    A
PROSEQ            EQ     SSP                                               PROSEQ     158    A
PROSEQ                                                                     PROSEQ     159    A
PROSEQ    SSP1    SA1    L.TXT       LT = L.TXT                            PROSEQ     160    A
PROSEQ            SA2    O.TXT       OT = O.TXT                            PROSEQ     161    A
PROSEQ            SA3    LM                                                PROSEQ     162    A
PROSEQ            LX7    X1                                                PROSEQ     163    A
PROSEQ            BX6    X3                                                PROSEQ     164    A
PROSEQ            SA7    L.SEQ       L.SEQ = L.TXT                         PROSEQ     165    A
PROSEQ            SA6    O.SEQ       O.SEQ = LM                            PROSEQ     166    A
PROSEQ            SA7    L.BLK       L.BLK = L.TXT                         PROSEQ     167    A
PROSEQ            SA6    O.BLK       O.BLK = LM                            PROSEQ     168    A
PROSEQ            SA7    X2+B1       TXTL[TXT+1] = L.TXT                   PROSEQ     169    A
PROSEQ            IX7    X3+X1                                             PROSEQ     170    A
PROSEQ            SA7    A2          O.TXT = LM + L.TXT                    PROSEQ     171    A
PROSEQ            SX6    4                                                 PROSEQ     172    A
PROSEQ            SA6    A1          L.TXT = 4                             PROSEQ     173    A
PROSEQ            MOVE   X1,X2,X3    MOVE( LT , OT , LM )                  PROSEQ     174    A
PROSEQ            EQ     SSP                                               PROSEQ     175    A
PROSEQ    TSB     TITLE  TSB - TRANSFER SYMBOL TABLE BITS TO DESCRIPTOR    PROSEQ     176    A
PROSEQ    *       PSB - TRANSFER L2 AND FP BITS FROM WORD B TO DESCRIPTOR. PROSEQ     177    A
PROSEQ    *       F.P. AND LEVEL 2 BITS.                                   PROSEQ     178    A
PROSEQ                                                                     PROSEQ     179    A
PROSEQ    PSB     ENTRY. **          **ENTRY/EXIT**                        PROSEQ     180         I
         -CCGNC
PROSEQ    PSB     ENTRY. **,#        **ENTRY/EXIT**                        CCGNC      728    A
PROSEQ            SA1    O.SYM                                             PROSEQ     181         I
         -CCG1
PROSEQ            SA1    =XO.SYM                                           CCG1       182    A
PROSEQ            SA5    O.TXT                                             PROSEQ     182    A
PROSEQ            SB4    4                                                 PROSEQ     183    A
PROSEQ            SA0    X1-1                                              PROSEQ     184         I
         -CCG1
PROSEQ            SA0    X1+1                                              CCG1       183    A
PROSEQ            SB5    I.GL                                              PROSEQ     185    A
PROSEQ            SB6    TSBA                                              PROSEQ     186    A
PROSEQ            SB3    B1                                                PROSEQ     187    A
PROSEQ            SX7    10B-T.LAB                                         PROSEQ     188         I
         -CCG1
PROSEQ            LX7    P.TYP      */  CARRY ADDEND FOR LABEL TEST        PROSEQ     189         I
         -CCG1
PROSEQ            ERRMI  10B-T.LAB        RECODE THIS SUBROUTINE           PROSEQ     190         I
         -CCG1
PROSEQ            SX7    B1                                                CCG1       184    A
PROSEQ            LX7    WB.LABP                                           CCG1       185    A
PROSEQ            MX0    -2                      */ MASK FOR F.P. AND LVL2 BITS  PROSEQ  191    A
PROSEQ            SA5    X5         R1 = O.TXT                             PROSEQ     192    A
PROSEQ                                                                     PROSEQ     193    A
PROSEQ    TSB1    SA5    A5+B4      R1 = R1 + 4                            PROSEQ     194    A
PROSEQ            SA4    A5+B1      R2 = R1 + 1                            PROSEQ     195    A
PROSEQ            ZR     B3,PSB     IF OC[R1-4] = 0                        PROSEQ     196    A
PROSEQ            UX6    B3,X5                                             PROSEQ     197    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ          LX4     -IH.IHP                                                 PROSEQ     198     A
PROSEQ          SB2     X4                                                      PROSEQ     199     A
PROSEQ          ZR      B2,TSB1     IF IH[R2] = 0                               PROSEQ     200     A
PROSEQ          GE      B2,B5,TSB1  IF I[R2] " 0                                PROSEQ     201     A
PROSEQ          SA2     A4+B1       DI = R2 + 1                                 PROSEQ     202     A
PROSEQ          SB2     B2+B2                                                   PROSEQ     203         I
         -CCG18
PROSEQ          LX2     59-D.TYP                                                PROSEQ     204     A
PROSEQ          MI      X2,TSB1     IF TYPE[DI] = II ! IV                       PROSEQ     205     A
PROSEQ          SA1     A0-B2       WORDB = [O.SYM-1-2*IH[R2]]                  PROSEQ     206         I
         -CCG1
PROSEQ          SX3     B2+B2                                                   CCG1       186     A
PROSEQ          SB2     A0+B2                                                   CCG1       187     A
PROSEQ          SA1     B2+X3       WORDB = SYM(3*IH+1)                         CCG1       188     A
PROSEQ          LX2     1+D.TYP                                                 PROSEQ     207     A
PROSEQ          IX5     X1+X7                                                   PROSEQ     208         I
         -CCG1
PROSEQ          MI      X5,TSB1     IF TYPE[WORDB] \ T.LAB  */ TRICY TEST       PROSEQ     209         I
         -CCG1
PROSEQ          BX5     X7*X1                                                   CCG1       189     A
PROSEQ          NZ      X5,TSB1     IF LAB[WORDB]                               CCG1       190     A
PROSEQ          LX1     -WB.FPP                                                 CCG1       191     A
PROSEQ          BX4     -X0*X1                                                  PROSEQ     210     A
PROSEQ          SA3     B6+X4       FP[DI] = FP[WORDB];  L2[DI] = L2[WORDB]     PROSEQ     211     A
PROSEQ          BX6     X2+X3                                                   PROSEQ     212         I
         -CCG35
PROSEQ          LX1     WB.FPP-WB.LEVNP                                 001390  CCG35       10     A
PROSEQ          MX4     -WB.LEVNL                                       001400  CCG35       11     A
PROSEQ          BX1     -X4*X1                                          001410  CCG35       12         I
         -CCGA014
PROSEQ          NZ      X1,TSB2     IF LEVN[WB] " 0 */ ^ LEVEL 0        001420  CCG35       13         I
         -CCGA014
PROSEQ          BX5     -X4*X1                                                  CCGA014      4     A
PROSEQ          NZ      X5,TSB2     IF LEVN[WB] NE 0  */ LEVEL 0                CCGA014      5     A
PROSEQ          SX5     B3-OC.STT                                       001430  CCG35       14     A
PROSEQ          ZR      X5,TSB2     IF OC[R1] = OC.STT                  001440  CCG35       15     A
PROSEQ          LX1     WB.LEVNP-WB.LCMP                                001450  CCG35       16     A
PROSEQ          SX6     B1                                              001460  CCG35       17     A
PROSEQ          BX1     X6*X1       1 IFF LCM[WORDB]                    001470  CCG35       18     A
PROSEQ          LX1     D.XMTP                                          001480  CCG35       19     A
PROSEQ          BX2     X1+X2       XMT[DI] = 1   */ INDICATE LEVEL 0   001490  CCG35       20     A
PROSEQ                                                                  001500  CCG35       21     A
PROSEQ   TSB2   BX6     X2+X3                                           001510  CCG35       22     A
PROSEQ          SA6     A2                                                      PROSEQ     213     A
PROSEQ          EQ      TSB1                                                    PROSEQ     214     A
PROSEQ                                                                          PROSEQ     215     A
PROSEQ   TSBA   BSS     0                                                       PROSEQ     216     A
PROSEQ          LOC     0                                                       PROSEQ     217     A
PROSEQ          CON     0                                                       PROSEQ     218     A
PROSEQ          BFMW    D,(FP)                                                  PROSEQ     219     A
PROSEQ          BFMW    D,(L2)                                                  PROSEQ     220     A
PROSEQ          BFMW    D,(FP,L2)                                               PROSEQ     221     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            BFMW   D,(FP,L2)                                                  PROSEQ      222     A
PROSEQ            LOC    *O                                                          PROSEQ      223     A
PROSEQ                                                                               CCG5        927     A
PROSEQ    .AET    ENDIF                                                               CCG5        928     A
PROSEQ    AIS     TITLE  AIS - ASSIGN INDEX REGISTERS IN STRAIGHT LINE CODE           PROSEQ      224     A
PROSEQ   **       AIS - ASSIGN INDEX REGISTERS IN STRAIGHT-LINE SEQUENCES.            PROSEQ      225     A
PROSEQ   *                                                                            PROSEQ      226     A
PROSEQ   *        AIS CHOOSES CANDIDATES FOR B-REGISTER USE IN STRAIGHT-LINE     AIS  PROSEQ      227     A
PROSEQ   *        SEQUENCES, WHEN THE REG WIDTH OVER ANY SUBSEQUENCE IS GREATER  AIS  PROSEQ      228     A
PROSEQ   *        THAN OR EQUAL TO TRW.  CANDIDATES ARE CHOSEN FROM *RF*         AIS  PROSEQ      229     A
PROSEQ   *        DEFINITIONS, I.E., THE PREDECESSORS OF TYPE III RF FIELDS AND  AIS  PROSEQ      230     A
PROSEQ   *        PLD/PST RJ FIELDS.                                             AIS  PROSEQ      231     A
PROSEQ   *        NO OPTIMAL PACKING IS ATTEMPTED;  BUT RATHER, A REGISTER IS    AIS  PROSEQ      232     A
PROSEQ   *        ASSIGNED TO EACH CANDIDATE SEQUENTIALLY, IF THAT CANDIDATE/S   AIS  PROSEQ      233     A
PROSEQ   *        EXTENDED USE SPAN (THE SPAN OF THE USES OF ALL USES) DOES NOT  AIS  PROSEQ      234     A
PROSEQ   *        INTERFERE WITH THAT REGISTER/S PREVIOUS ASSIGNMENT OR ANY LOCK AIS  PROSEQ      235     A
PROSEQ   *        TO THAT REGISTER THAT MAY BE INCOMING WITH THE SEQUENCE.       AIS  PROSEQ      236     A
PROSEQ   *                                                                            PROSEQ      237     A
PROSEQ   *        TRACER (AIS,AIMOD,AIMPB,AIENT,AILBM)                                PROSEQ      238     A
PROSEQ                                                                               PROSEQ      239     A
PROSEQ            QUAL   AIS                                                          PROSEQ      240     A
PROSEQ                                                                               PROSEQ      241     A
PROSEQ    TRW     MDV=   7,7,6,6     THRESHHOLD REGISTER WIDTH                        PROSEQ      242         I
         -F2950CC
PROSEQ            IFLT   .CPU,75                                                      F2950CC      10     A
PROSEQ    TRW     EQU    7                                                            F2950CC      11     A
PROSEQ            ELSE                                                                F2950CC      12     A
PROSEQ    TRW     EQU    6                                                            F2950CC      13     A
PROSEQ            ENDIF                                                               F2950CC      14     A
PROSEQ                                                                        .NO.ID. PROSEQ      243     A
PROSEQ   *        EQUATE THREE TEMP TABLES.                                   .NO.ID. PROSEQ      244         I
         -CCG16
PROSEQ                                                                        .NO.ID. PROSEQ      245         I
         -CCG16
PROSEQ            RMT                                                            AIS  PROSEQ      246         I
         -CCG5    -CCG16
PROSEQ            ECHO   ,A=(BCT,DUT,PBT),B=(RCT,RND,PIT)                        AIS  PROSEQ      247         I
         -CCG16
PROSEQ            EXT    L.B,O.B,Z.B                                                  CCG5        929         I
         -CCGNC   -CCG16
PROSEQ            EXT    Z$B                                                          CCGNC       729         I
         -CCG16
PROSEQ    Z$A     EQU    Z$B                                                          CCGNC       730         I
         -CCG16
PROSEQ    L.A     EQU    L.B                                                    AIS  PROSEQ      248         I
         -CCG16
PROSEQ    O.A     EQU    O.B                                                    AIS  PROSEQ      249         I
         -CCG16
PROSEQ    Z.A     EQU    Z.B                                                    AIS  PROSEQ      250         I
         -CCGNC   -CCG16
PROSEQ            ENDD                                                           AIS  PROSEQ      251         I
         -CCG16

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            TABLES (BCT,DUT,PBT)                                          CCG16        244     A
PROSEQ            RMT                                                  AIS      PROSEQ       252         I
         -CCG5
PROSEQ                                                                 AIS      PROSEQ       253     A
PROSEQ    .RTP    BIT    SO.RTP                                        AIS      PROSEQ       254         I
         -FCC0641
PROSEQ    SO.BR   EQU    RT.BR*.RTP                                    AIS      PROSEQ       255         I
         -FCC0641
PROSEQ    SO.BLOCK EQU   SO.LOCK+SO.BR                                 AIS      PROSEQ       256         I
         -FCC0641
PROSEQ    .SFL    BIT    SO.SFLP                                               FCC0641       15     A
PROSEQ    SO.BLOCK EQU   SO.LOCK+.SFL      SCRATACHABLE FULL B-REG LOCK        FCC0641       16     A
PROSEQ                                                                 AIS      PROSEQ       257     A
PROSEQ    LU.INF  EQU    1S17-1      INFINITE LAST USE                 AIS      PROSEQ       258     A
PROSEQ                                                                 AIS      PROSEQ       259     A
PROSEQ    RIL     BIT    R1.RIL                                        .NO.ID.  PROSEQ       260         I
         -CCG11
PROSEQ    RMSK    EQU    RIL-1-1                                       .NO.ID.  PROSEQ       261         I
         -CCG11
PROSEQ    RMSK    EQU    -RN.MASK                                              CCG11        123     A
PROSEQ                                                                 AIS      PROSEQ       262     A
PROSEQ                                                                 AIS      PROSEQ       263     A
PROSEQ   *        BOUNDARY MARKING INSTRUCTION LINK WORD FORMAT.       AIS      PROSEQ       264     A
PROSEQ   *        THE LINK WORDS OF SUBSEQUENCE DEMARCATING INSTRUCTIONS ARE      PROSEQ       265     A
PROSEQ   *        LINKED DURING *LBM*, AND CONTAIN INFORMATION REGARDING THE      PROSEQ       266     A
PROSEQ   *        PROFITIBILITY OF FURTHER PROCESSING THIS SUBSEQUENCE.           PROSEQ       267     A
PROSEQ                                                                 AIS      PROSEQ       268     A
PROSEQ            DESCRIBE  BM.,60   BMW(MRW,NRF,P)                    AIS      PROSEQ       269     A
PROSEQ            DEFINE 6                                             AIS      PROSEQ       270     A
PROSEQ    MRW     DEFINE 18          MAXIMUM REG WIDTH ACROSS SUB-SEQUENCE AIS      PROSEQ       271     A
PROSEQ    NRF     DEFINE 18          NUMBER OF *RF* DEFINITIONS IN SUBSEQUENCE AIS  PROSEQ       272     A
PROSEQ    P       DEFINE 18          LINK TO NEXT BM WORD                 AIS      PROSEQ       273     A
PROSEQ                                                                 AIS      PROSEQ       274     A
PROSEQ   *        BCT - B REGISTER CANDIDATE TABLE ENTRY FORMAT.       AIS      PROSEQ       275     A
PROSEQ                                                                 AIS      PROSEQ       276     A
PROSEQ            DESCRIBE  BC.,60   BCW(MRW,LXU,RI)                   AIS      PROSEQ       277     A
PROSEQ            DEFINE 6                                             AIS      PROSEQ       278     A
PROSEQ    MRW     DEFINE 18          MAXIMUM REGISTER WIDTH ACROSS EXT. USE SPAN AIS PROSEQ  279     A
PROSEQ    LXU     DEFINE 18          INDEX IN *SEQ* OF LAST EXTENDED USE  AIS      PROSEQ       280     A
PROSEQ    RI      DEFINE 18          INDEX IN SEQ OF DEFINITION OF THIS CAND AIS    PROSEQ       281     A
PROSEQ                                                                 AIS      PROSEQ       282     A
PROSEQ   *        PBT - PREASSIGNED B-REGISTER TABLE ENTRY FORMAT.     AIS      PROSEQ       283     A
PROSEQ                                                                 AIS      PROSEQ       284     A
PROSEQ            DESCRIBE  PB.,60   PBW(RN,RI,LU,P)                   AIS      PROSEQ       285     A
PROSEQ            DEFINE 3                                             AIS      PROSEQ       286     A
PROSEQ    RN      DEFINE 3           B-REGISTER NUMBER OF THIS LOCK       AIS      PROSEQ       287     A
PROSEQ    RI      DEFINE 18          INDEX IN *SEQ* OF THIS DEFINITION    AIS      PROSEQ       288     A
PROSEQ    LU      DEFINE 18          LAST USE OF THIS LOCK                AIS      PROSEQ       289     A
PROSEQ    P       DEFINE 18          LINK TO NEXT PBT ENTRY OF SAME RN    AIS      PROSEQ       290     A
PROSEQ                                                                 AIS      PROSEQ       291     A
PROSEQ   *        DUT - DEF-USE TABLE ENTRY FORMAT.                    AIS      PROSEQ       292     A
PROSEQ                                                                 AIS      PROSEQ       293     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            DESCRIBE DU.,60    DUW(DEF,USE)                                  AIS     PROSEQ       294     A
PROSEQ    SU      DEFINE 1           SHORT USE                                     AIS     PROSEQ       295     A
PROSEQ    SC      DEFINE 11          SHIFT COUNT TO MOVE R-FIELD USED TO BIT 0     AIS     PROSEQ       296     A
PROSEQ            DEFINE 12                                                        AIS     PROSEQ       297     A
PROSEQ    DEF     DEFINE 18          RI OF DEFINITION                              AIS     PROSEQ       298     A
PROSEQ    USE     DEFINE 18          INDEX IN *SEQ* OF USE                         AIS     PROSEQ       299     A
PROSEQ                                                                             AIS     PROSEQ       300     A
PROSEQ    *       CAT - CURRENT ASSIGNMENT TABLE ENTRY FORMAT.                     AIS     PROSEQ       301     A
PROSEQ                                                                             AIS     PROSEQ       302     A
PROSEQ            DESCRIBE  CA.,60                                                 AIS     PROSEQ       303     A
PROSEQ            DEFINE 6                                                         AIS     PROSEQ       304     A
PROSEQ            DEFINE 18                                                        AIS     PROSEQ       305     A
PROSEQ    LXU     DEFINE 18                                                        AIS     PROSEQ       306     A
PROSEQ    RI      DEFINE 18                                                        AIS     PROSEQ       307     A
PROSEQ                                                                             AIS     PROSEQ       308     A
PROSEQ    *       FORMAT OF LINK WORD AFTER BUILDING OF DUT.                       AIS     PROSEQ       309     A
PROSEQ                                                                             AIS     PROSEQ       310     A
PROSEQ            DESCRIBE  DP.,60                                                 AIS     PROSEQ       311     A
PROSEQ            DEFINE 24                                                        AIS     PROSEQ       312     A
PROSEQ    II      DEFINE 18          POINTS TO DUT ENTRY FOR FIRST USE             AIS     PROSEQ       313     A
PROSEQ    NI      DEFINE 18          POINTS TO NEXT DUT ENTRY FOR THIS RI          AIS     PROSEQ       314     A
PROSEQ                                                                             AIS     PROSEQ       315     A
PROSEQ    *       PBI - PREASSIGNED B-REG LOCK INDEX TABLE ENTRY FORMAT.           AIS     PROSEQ       316     A
PROSEQ                                                                             AIS     PROSEQ       317     A
PROSEQ            DESCRIBE  PI.,60                                                 AIS     PROSEQ       318     A
PROSEQ    NA      DEFINE 1           THIS REG OFF MARKET FOR REST OF *SEQ*         AIS     PROSEQ       319     A
PROSEQ            DEFINE 41                                                        AIS     PROSEQ       320     A
PROSEQ    P       DEFINE 18          LINK TO NEXT PERTINENT PBT ENTRY              .NO.ID. PROSEQ       321     A
PROSEQ                                                                             AIS     PROSEQ       322     A
PROSEQ    *       FIXED STORAGE DECLARATIONS.                                      .NO.ID. PROSEQ       323     A
PROSEQ                                                                             AIS     PROSEQ       324     A
PROSEQ            USE    /TABLES/                                                  AIS     PROSEQ       325         I
          -CCG1
PROSEQ            USE    /CCG.SCR/                                                         CCG1         192         I
          -CCG18
PROSEQ            USE    /CCGSCR/                                                          CCG18        112     A
PROSEQ    FLF     BSS    1           FULL LOCK FLAGS                                   PROSEQ       326     A
PROSEQ    PBI     BSS    7           PBT INDEX VECTOR                              AIS     PROSEQ       327     A
PROSEQ    CAT     BSS    7           CURRENT ASSIGNMENT TABLE                      AIS     PROSEQ       328     A
PROSEQ    NSO     BSS    1           NEXT SEQUENCE ORIGIN INDEX                    AIS     PROSEQ       329     A
PROSEQ    MRF     BSS    1           MAX(RF DEFS IN SUBSEQUENCE)                   .NO.ID. PROSEQ       330     A
PROSEQ    LBMA    BSS    1           SAVE (O.SEQ-2) IN LBM                         .NO.ID. PROSEQ       331     A
PROSEQ    IUMB    BSS    1           SAVE (UL) IN IUM                              .NO.ID. PROSEQ       332     A
PROSEQ            USE    0                                                         AIS     PROSEQ       333     A
PROSEQ    AIS     EJECT                                                            AIS     PROSEQ       334     A
PROSEQ    **      AIS - MAIN LOOP                                                  AIS     PROSEQ       335     A
PROSEQ                                                                             AIS     PROSEQ       336     A
PROSEQ    AIS0    SX6    0                                                         AIS     PROSEQ       337     A
PROSEQ            SA6    L.BCT       L.BCT = 0                                     AIS     PROSEQ       338     A
PROSEQ            SA6    L.DUT       L.DUT = 0                                     AIS     PROSEQ       339     A
PROSEQ            SA6    L.PBT       L.PBT = 0                                     AIS     PROSEQ       340     A
PROSEQ                                                                             AIS     PROSEQ       341     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            TRACE  AIMOD,RLIST,MOD                                          PROSEQ     342    A
PROSEQ            TRACE  AIMOD,MLT                                                PROSEQ     343    A
PROSEQ            CALL   MPB         MERGE MODS WITH SEQUENCE (NO SQUEEZE)   AIS  PROSEQ     344         I
         -CCGNC
PROSEQ            CALL   MPB#        MERGE MODS WITH SEQUENCE ( NO SQZ )         CCGNC      731    A
PROSEQ            TRACE  AIMPB,RLIST                                              PROSEQ     345    A
PROSEQ                                                                       AIS  PROSEQ     346    A
PROSEQ            CALL   CUC         COLLECT USES COUNTS                     AIS  PROSEQ     347         I
         -CCGNC
PROSEQ            CALL   CUC#        COLLECT USES COUNTS                         CCGNC      732    A
PROSEQ            TRACE  AIS,RLIST                                             AIS  PROSEQ     348    A
PROSEQ                                                                       AIS  PROSEQ     349    A
PROSEQ    AIS     ENTRY. **          **ENTRY/EXIT**                          AIS  PROSEQ     350         I
         -CCGNC
PROSEQ    AIS     ROUTINE                                                         CCGNC      733    A
PROSEQ            SA1    O.SEQ                                                 AIS  PROSEQ     351    A
PROSEQ            SA2    X1+2                                                  AIS  PROSEQ     352    A
PROSEQ            MX0    -D.REGWL                                              AIS  PROSEQ     353    A
PROSEQ            LX2    -D.REGWP                                              AIS  PROSEQ     354    A
PROSEQ            BX6    -X0*X2                                                AIS  PROSEQ     355    A
PROSEQ            SX0    X6-TRW                                                AIS  PROSEQ     356    A
PROSEQ            TRACE  AIENT,RLIST,SEQ                                          PROSEQ     357    A
PROSEQ                                                                       AIS  PROSEQ     358    A
PROSEQ            MI     X0,AIS0     IF MAXW .LT. TRW                        AIS  PROSEQ     359    A
PROSEQ            RJ     LBM         LOCATE SEQUENCE BOUNDARIES              AIS  PROSEQ     360    A
PROSEQ            TRACE  AILBM,RLIST,SEQ                                          PROSEQ     361    A
PROSEQ                                                                       AIS  PROSEQ     362    A
PROSEQ            SA1    MRF                                                   AIS  PROSEQ     363    A
PROSEQ            SX5    A0                                                    AIS  PROSEQ     364    A
PROSEQ            ZR     X1,AIS0     IF NO RF USES IN SEQ                    AIS  PROSEQ     365    A
PROSEQ                                                                       AIS  PROSEQ     366    A
PROSEQ    *       ALLOCATE FOR TABLE OF ALL POSSIBLE B-ASSIGNMENTS.            AIS  PROSEQ     367    A
PROSEQ                                                                       AIS  PROSEQ     368    A
PROSEQ            ALLOC  BCT,B1+X1   ALLOC (BCT,MRF+1)                       AIS  PROSEQ     369    A
PROSEQ                                                                       AIS  PROSEQ     370    A
PROSEQ            BX1    X5                                                    AIS  PROSEQ     371    A
PROSEQ            LX5    2                                                     AIS  PROSEQ     372    A
PROSEQ            ALLOC  MLT,B1+X1   ALLOC (MLT,MMOD+1)                      AIS  PROSEQ     373    A
PROSEQ                                                                       AIS  PROSEQ     374    A
PROSEQ            ALLOC  MOD,X5+4    ALLOC(MOD,4*(MMOD+1))                   AIS  PROSEQ     375    A
PROSEQ                                                                       AIS  PROSEQ     376    A
PROSEQ            SA1    L.SEQ                                                 AIS  PROSEQ     377    A
PROSEQ            AX2    B1,X1                                                 AIS  PROSEQ     378    A
PROSEQ            AX5    B1,X2                                                 AIS  PROSEQ     379    A
PROSEQ            IX1    X2+X5                                                 AIS  PROSEQ     380    A
PROSEQ                                                                       AIS  PROSEQ     381    A
PROSEQ    *       ALLOCATE FOR DEF-USE *TREE*.                                 AIS  PROSEQ     382    A
PROSEQ                                                                       AIS  PROSEQ     383    A
PROSEQ            ALLOC  DUT,X1      ALLOC (DUT,3/4*L.SEQ)                   AIS  PROSEQ     384    A
PROSEQ                                                                       AIS  PROSEQ     385    A
PROSEQ    *       ALLOCATE FOR TABLE OF LOCKS TO B-REGS INCOMING WITH SEQUENCE. AIS  PROSEQ     386    A
PROSEQ                                                                       AIS  PROSEQ     387    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            ALLOC  PBT,X5      ALLOC (PBT,1/4*L.SEQ)                AIS     PROSEQ     388    A
PROSEQ                                                                   AIS     PROSEQ     389    A
PROSEQ            SX7    4                                               AIS     PROSEQ     390    A
PROSEQ            MX6    0                                               AIS     PROSEQ     391    A
PROSEQ            SA7    L.MOD       L.MOD = 4                           AIS     PROSEQ     392    A
PROSEQ            SA6    L.MLT       L.MLT = 0                           AIS     PROSEQ     393    A
PROSEQ            SA1    O.SEQ                                           AIS     PROSEQ     394    A
PROSEQ            SB2    X1+4        R1F = O.SEQ + 4                     AIS     PROSEQ     395    A
PROSEQ            SA0    X1                                              AIS     PROSEQ     396    A
PROSEQ                                                                   .NO.ID. PROSEQ     397    A
PROSEQ   *        A0 CONTAINS O.SEQ THROUGHOUT AIS.                      .NO.ID. PROSEQ     398    A
PROSEQ                                                                   .NO.ID. PROSEQ     399    A
PROSEQ            SA2    B2-B1       LI = R1F - 1 ;  LIW = [LI]          AIS     PROSEQ     400    A
PROSEQ            SA6    FLF         FLF = 0                                     PROSEQ     401    A
PROSEQ            EQ     AIS2                                            AIS     PROSEQ     402    A
PROSEQ                                                                   .NO.ID. PROSEQ     403    A
PROSEQ   *        PROCESS ALL SUBSEQUENCES, AS DEMARCATED BY *LBM*.      .NO.ID. PROSEQ     404    A
PROSEQ                                                                   AIS     PROSEQ     405    A
PROSEQ    AIS1    SA1    NSO                                             AIS     PROSEQ     406    A
PROSEQ            LX1    -BM.PP                                          AIS     PROSEQ     407    A
PROSEQ            SB4    X1                                              AIS     PROSEQ     408    A
PROSEQ            SX1    A0+B4       LI = P[NSO] + O.SEQ                 AIS     PROSEQ     409    A
PROSEQ            SB2    X1+B1       R1F = LI + 1                        AIS     PROSEQ     410    A
PROSEQ            SA2    X1          LIW = [LI]                          AIS     PROSEQ     411    A
PROSEQ                                                                   AIS     PROSEQ     412    A
PROSEQ    AIS2    SB3    X2                                              AIS     PROSEQ     413    A
PROSEQ            ZR     B3,AIS0     IF P[LIW] = 0  */ END OF SEQ        AIS     PROSEQ     414    A
PROSEQ            BX6    X2                                              AIS     PROSEQ     415    A
PROSEQ            LX2    -BM.PP                                          AIS     PROSEQ     416    A
PROSEQ            SB3    X2-3        R1L = P[LIW] - 3                    AIS     PROSEQ     417    A
PROSEQ            SB3    A0+B3                                           AIS     PROSEQ     418    A
PROSEQ            SA6    NSO         [NSO] = LIW                         AIS     PROSEQ     419    A
PROSEQ            LE     B3,B2,AIS1  IF R1L .LE. R1F                     AIS     PROSEQ     420    A
PROSEQ            LX6    -BM.NRFP                                        AIS     PROSEQ     421    A
PROSEQ            SX4    X6                                              AIS     PROSEQ     422    A
PROSEQ            ZR     X4,AIS1     IF NRF[LIW] = 0                     AIS     PROSEQ     423    A
PROSEQ            LX6    BM.NRFP-BM.MRWP                                 AIS     PROSEQ     424    A
PROSEQ            SX5    X6-TRW                                          AIS     PROSEQ     425    A
PROSEQ            MI     X5,AIS1     IF MRW[LIW] .LT. TRW                AIS     PROSEQ     426    A
PROSEQ                                                                   AIS     PROSEQ     427    A
PROSEQ   *        TRANSFER FULL LOCKS PREASSIGNED IN PREVIOUS SUBSEQUENCE TO  .NO.ID. PROSEQ 428    A
PROSEQ   *        PBI.  CLEAR PBI LINK FIELD.                            .NO.ID. PROSEQ     429    A
PROSEQ                                                                   AIS     PROSEQ     430    A
PROSEQ            SA1    FLF                                             AIS     PROSEQ     431    A
PROSEQ            SB4    59                                              AIS     PROSEQ     432    A
PROSEQ            MX7    0                                               AIS     PROSEQ     433    A
PROSEQ            SB5    B0          BI = 0                              AIS     PROSEQ     434    A
PROSEQ            SB6    7                                               .NO.ID. PROSEQ     435    A
PROSEQ                                                                   AIS     PROSEQ     436    A
PROSEQ    AIS3    LX1    -1                                              .NO.ID. PROSEQ     437    A
PROSEQ            SB5    B5+1        BI = BI + 1                         .NO.ID. PROSEQ     438    A
PROSEQ            AX6    B4,X1       PBI(BI) = 0                         AIS     PROSEQ     439    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            SA6    PBI-1+B5    NA[PBI(BI)] = FLF(BI)                      AIS     PROSEQ     440    A
PROSEQ            NE     B5,B6,AIS3  IF BI .NE. 7                               .NO.ID. PROSEQ     441    A
PROSEQ                                                                          AIS     PROSEQ     442    A
PROSEQ    AIS4    SA7    A1          FLF = 0                                    AIS     PROSEQ     443    A
PROSEQ            SA1    O.PBT                                                  AIS     PROSEQ     444    A
PROSEQ            SX6    B1                                                     AIS     PROSEQ     445    A
PROSEQ            SA7    L.BCT       L.BCT = 0                                  AIS     PROSEQ     446    A
PROSEQ            SA7    X1+         [O.PBT] = 0                                AIS     PROSEQ     447    A
PROSEQ            SA6    L.PBT       L.PBT = 1                                  AIS     PROSEQ     448    A
PROSEQ            RJ     BDU         BUILD DEF-USE TABLE AND BCT AND PBT        AIS     PROSEQ     449    A
PROSEQ                                                                          AIS     PROSEQ     450    A
PROSEQ    .T      IFNE   TEST,0                                                 .NO.ID. PROSEQ     451    A
PROSEQ            SA2    O.DUT                                                  AIS     PROSEQ     452    A
PROSEQ            SA1    B3-B1                                                  AIS     PROSEQ     453    A
PROSEQ            LX1    -DP.NIP                                                AIS     PROSEQ     454    A
PROSEQ            IX5    X1-X2                                                  AIS     PROSEQ     455    A
PROSEQ            SX6    X5+                                                    AIS     PROSEQ     456    A
PROSEQ            SA6    L.DUT       L.DUT = NI[R1L-1] - O.DUT                  AIS     PROSEQ     457    A
PROSEQ    .T      ENDIF                                                         .NO.ID. PROSEQ     458    A
PROSEQ                                                                          .NO.ID. PROSEQ     459    A
PROSEQ            TRACE  AIS,RLIST,SEQ                                          AIS     PROSEQ     460    A
PROSEQ            TRACE  AIS,(DUT,BCT,PBT)                                      AIS     PROSEQ     461    A
PROSEQ                                                                          AIS     PROSEQ     462    A
PROSEQ            RJ     IXU         INSERT EXTENDED USE INFO INTO BCT          AIS     PROSEQ     463    A
PROSEQ            TRACE  AIS,BCT                                                AIS     PROSEQ     464    A
PROSEQ                                                                          AIS     PROSEQ     465    A
PROSEQ            SX5    X4-TRW                                                 AIS     PROSEQ     466    A
PROSEQ            MI     X5,AIS1     IF SMAXW .LT. TRW                          AIS     PROSEQ     467    A
PROSEQ                                                                          AIS     PROSEQ     468    A
PROSEQ            RJ     IUP         INSERT USE INFO INTO PBT                   AIS     PROSEQ     469    A
PROSEQ            TRACE  AIS,PBT                                                AIS     PROSEQ     470    A
PROSEQ                                                                          AIS     PROSEQ     471    A
PROSEQ            RJ     ABR         ASSIGN B-REGS AND TABLE MODS               AIS     PROSEQ     472    A
PROSEQ            TRACE  AIS,RLIST,MOD                                          AIS     PROSEQ     473    A
PROSEQ            TRACE  AIS,MLT                                                AIS     PROSEQ     474    A
PROSEQ            EQ     AIS1                                                   AIS     PROSEQ     475    A
PROSEQ     LBM    EJECT               LBM                                                PROSEQ     476    A
PROSEQ    **      LBM - LOCATE BOUNDARY MARKERS.                                AIS     PROSEQ     477    A
PROSEQ    *       THIS ROUTINE CHAINS ALL SEQUENCE-BREAKING INSTRUCTIONS THROUGH AIS     PROSEQ     478    A
PROSEQ    *       THE LINK WORDS, BEGINNING WITH THE *BOS*.  IT SIMULTANEOUSLY  AIS     PROSEQ     479    A
PROSEQ    *       INSTALLS THE MAXIMUM REG WIDTH OVER AND THE NUMBER OF *RF*    AIS     PROSEQ     480    A
PROSEQ    *       DEFINITIONS IN THE SEQUENCE INTO THE BOUNDARY MARKER LINKS.   AIS     PROSEQ     481    A
PROSEQ    *       LARGEST POSSIBLE *MOD* SIZE IS ALSO CALCULATED DURING THIS    AIS     PROSEQ     482    A
PROSEQ    *       BACKWARD SCAN.                                                AIS     PROSEQ     483    A
PROSEQ    *                                                                     AIS     PROSEQ     484    A
PROSEQ    *       EXIT   MRF - CONTAINS THE MAXIMUM NUMBER OF *RF* DEFS IN ANY  AIS     PROSEQ     485    A
PROSEQ    *              SUBSEQUENCE.                                           AIS     PROSEQ     486    A
PROSEQ    *              A0 - MMOD, OVERESTIMATE OF POSSIBLE *MOD* ENTRIES.     AIS     PROSEQ     487    A
PROSEQ                                                                          AIS     PROSEQ     488    A
PROSEQ     LBM0   SA1    LBMA                                                   AIS     PROSEQ     489    A
PROSEQ            SA2    O.SEQ                                                  AIS     PROSEQ     490    A
PROSEQ            BX6    X1                                                     AIS     PROSEQ     491    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            SA6    X2-2        [O.SEQ-2] = SAVE                          AIS    PROSEQ    492    A
PROSEQ                                                                         AIS    PROSEQ    493    A
PROSEQ    LBM     ROUTINE                                                      AIS    PROSEQ    494    A
PROSEQ            SA1    O.SEQ                                                 AIS    PROSEQ    495    A
PROSEQ            SA3    X1+B1                                                 AIS    PROSEQ    496    A
PROSEQ            SB2    X3-2        LD = TXTL - 2                             AIS    PROSEQ    497    A
PROSEQ            SA5    X1+B2       I = N.INS ; DIN = O.SEQ + LD ; DINW = [DIN] AIS  PROSEQ    498    A
PROSEQ            SA3    X1-2                                                  AIS    PROSEQ    499    A
PROSEQ            MX7    0                                                     AIS    PROSEQ    500    A
PROSEQ            BX6    X3          SAVE = [O.SEQ-2]                          AIS    PROSEQ    501    A
PROSEQ            SA7    A3          [O.SEQ-2] = 0                             AIS    PROSEQ    502    A
PROSEQ            SA6    LBMA                                                  AIS    PROSEQ    503    A
PROSEQ            BX1    X7          TUSES = 0                                 AIS    PROSEQ    504    A
PROSEQ            MX2    0           MAXW = 0                                  AIS    PROSEQ    505    A
PROSEQ            SA0    B0          MMOD = 0                                  AIS    PROSEQ    506    A
PROSEQ            SB4    B0          PLINK = 0                                 AIS    PROSEQ    507    A
PROSEQ            SB3    B0          NRF = 0                                   AIS    PROSEQ    508    A
PROSEQ            SA7    MRF         MRF = 0                                   AIS    PROSEQ    509    A
PROSEQ            EQ     LBM5                                                  AIS    PROSEQ    510    A
PROSEQ                                                                         AIS    PROSEQ    511    A
PROSEQ    *       BOUNDARY MARKER BREAKS SUBSEQUENCE.  FILE BM WORD FOR SEQUENCE AIS  PROSEQ    512    A
PROSEQ    *       JUST SCANNED (FOLLOWING).                                    AIS    PROSEQ    513    A
PROSEQ                                                                         AIS    PROSEQ    514    A
PROSEQ    LBM1    LX6    D.BMP-D.TYP-1                                         AIS    PROSEQ    515    A
PROSEQ            IX7    X6+X6                                                 AIS    PROSEQ    516    A
PROSEQ            BX4    -X7*X6                                                AIS    PROSEQ    517    A
PROSEQ            BX0    -X6+X7                                                AIS    PROSEQ    518    A
PROSEQ            MI     X4,LBM5     IF TY[DIW] = II                           AIS    PROSEQ    519    A
PROSEQ            SA3    A5+2        R1I = DIN + 2 ;  RIW = [R1I]                     PROSEQ    520    A
PROSEQ            UX4    B2,X3                                                        PROSEQ    521    A
PROSEQ            SB5    B2-OC.EOS                                                    PROSEQ    522    A
PROSEQ            NZ     B5,LBM1A    IF OC[R1W] .NE. OC.EOS                           PROSEQ    523    A
PROSEQ            SX4    X3                                                           PROSEQ    524    A
PROSEQ            NZ     X4,LBM5     IF IH[RIW] .NE. 0   */ OPT=2 EOS                 PROSEQ    525    A
PROSEQ                                                                               PROSEQ    526    A
PROSEQ    LBM1A   MI     X0,LBM2     IF TY[DIW] .NE. III                              PROSEQ    527    A
PROSEQ            SB5    B2-OC.JIN                                             AIS    PROSEQ    528    A
PROSEQ            NZ     B5,LBM5     IF OC[RIW] .NE. OC.JIN                    AIS    PROSEQ    529    A
PROSEQ                                                                         AIS    PROSEQ    530    A
PROSEQ    LBM2    SX6    B3                                                    AIS    PROSEQ    531    A
PROSEQ            SA3    MRF                                                   AIS    PROSEQ    532    A
PROSEQ            SB7    B3+B3                                                 AIS    PROSEQ    533    A
PROSEQ            IX7    X3-X6                                                 AIS    PROSEQ    534    A
PROSEQ            SB7    B7+B3                                                 AIS    PROSEQ    535    A
PROSEQ            SB3    B0          NRF = 0                                   AIS    PROSEQ    536    A
PROSEQ            PL     X7,LBM3     IF MRF .GE. NRF                           AIS    PROSEQ    537    A
PROSEQ            SA6    A3          MRF = NRF                                 AIS    PROSEQ    538    A
PROSEQ                                                                         AIS    PROSEQ    539    A
PROSEQ    LBM3    LX2    BM.MRWP                                               AIS    PROSEQ    540    A
PROSEQ            SB5    B7+X1                                                 AIS    PROSEQ    541    A
PROSEQ            SX7    B4                                                    AIS    PROSEQ    542    A
PROSEQ                                                                         AIS    PROSEQ    543    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ   *        FOR EACH *RF* DEF WE WILL ALLOCATE *MOD* FOR ALL USES PLUS     .NO.ID. PROSEQ    544   A
PROSEQ   *        3 (FOR THE DEF, SA AND RS).                                    .NO.ID. PROSEQ    545   A
PROSEQ                                                                           AIS     PROSEQ    546   A
PROSEQ            SA0    A0+B5       MMOD = MMOD + TUSES + 3*NRF                 AIS     PROSEQ    547   A
PROSEQ            LX6    BM.NRFP                                                 AIS     PROSEQ    548   A
PROSEQ            BX2    X6+X2                                                   AIS     PROSEQ    549   A
PROSEQ            LX7    BM.PP                                                   AIS     PROSEQ    550   A
PROSEQ            BX6    X7+X2                                                   AIS     PROSEQ    551   A
PROSEQ            SA1    O.SEQ                                                   AIS     PROSEQ    552   A
PROSEQ            SA6    A5+5        LI = DIN + 5 ;  [LI] = BMW(NRF,PLINK,MAXW)  AIS     PROSEQ    553   A
PROSEQ            SB5    X1                                                      AIS     PROSEQ    554   A
PROSEQ            MX2    0           MAXW = 0                                    AIS     PROSEQ    555   A
PROSEQ            SB4    A6-B5       PLINK = LI - O.SEQ                          AIS     PROSEQ    556   A
PROSEQ            MX1    0           USES = 0                                    AIS     PROSEQ    557   A
PROSEQ            ZR     X5,LBM0     IF DINW = 0   */ IF I = 0                   AIS     PROSEQ    558   A
PROSEQ            EQ     LBM5                                                    AIS     PROSEQ    559   A
PROSEQ                                                                           AIS     PROSEQ    560   A
PROSEQ    LBM4    LX6    D.RFP-D.BMP                                             AIS     PROSEQ    561   A
PROSEQ            BX7    X6                                                              PROSEQ    562   A
PROSEQ            ZR     X5,LBM1     IF DINW = 0   */ BOS                        AIS     PROSEQ    563   A
PROSEQ            LX7    D.BMP-D.^DP                                                     PROSEQ    564   A
PROSEQ            PL     X7,LBM5     IF D[DIW]     */ POSSIBLE DEF                       PROSEQ    565   A
PROSEQ            MI     X6,LBM1     IF BM[DIW]                                  AIS     PROSEQ    566   A
PROSEQ                                                                           AIS     PROSEQ    567   A
PROSEQ    LBM5    BX6    X5          DIW = DINW                                  AIS     PROSEQ    568   A
PROSEQ            SA5    A5-4        I = I - 1 ;  DIN = DIN - 4 ;  DINW = [DIN]  AIS     PROSEQ    569   A
PROSEQ            MX0    -D.REGWL                                                AIS     PROSEQ    570   A
PROSEQ            LX6    -D.REGWP                                                AIS     PROSEQ    571   A
PROSEQ            BX4    -X0*X6      REG = REGW[DIW]                             AIS     PROSEQ    572   A
PROSEQ            IX7    X2-X4                                                   AIS     PROSEQ    573   A
PROSEQ            PL     X7,LBM6     IF MAXW .GE. REG                            AIS     PROSEQ    574   A
PROSEQ            BX2    X4          MAXW = REG                                  AIS     PROSEQ    575   A
PROSEQ                                                                           AIS     PROSEQ    576   A
PROSEQ    LBM6    LX6    59-D.RFP+D.REGWP                                        AIS     PROSEQ    577   A
PROSEQ            PL     X6,LBM4     IF ^RF[DIW]                                 AIS     PROSEQ    578   A
PROSEQ                                                                           AIS     PROSEQ    579   A
PROSEQ   *        *RF* DEFINITION FOUND.  CONSIDER AS POSSIBLE CANDIDATE.        AIS     PROSEQ    580   A
PROSEQ                                                                           AIS     PROSEQ    581   A
PROSEQ            LX6    1+D.RFP-D.USESP                                         AIS     PROSEQ    582   A
PROSEQ            MX0    -D.USESL                                                AIS     PROSEQ    583   A
PROSEQ            SB3    B3+B1       NRF = NRF + 1                               AIS     PROSEQ    584   A
PROSEQ            BX4    -X0*X6                                                  AIS     PROSEQ    585   A
PROSEQ            IX1    X1+X4       TUSES = TUSES + USES[DIW]                   AIS     PROSEQ    586   A
PROSEQ            LX6    D.USESP+59-D.RFP                                        AIS     PROSEQ    587   A
PROSEQ            EQ     LBM4                                                    AIS     PROSEQ    588   A
PROSEQ    BDU     EJECT              BDU                                                  PROSEQ    589   A
PROSEQ   **       BDU - BUILD DEF-USE TABLE.                                     AIS     PROSEQ    590   A
PROSEQ   *        PERFORM FORWARD SCAN TO CONSTRUCT DUT FOR SUBSEQUENCE.         .NO.ID. PROSEQ    591   A
PROSEQ   *        THE BCT IS ALSO FORMED, OF *RF* DEFINITIONS.  FINALLY, THE PBT AIS     PROSEQ    592   A
PROSEQ   *        IS FORMED OF B-REG ASSIGNMENTS INCOMING WITH THE SEQUENCE.     AIS     PROSEQ    593   A
PROSEQ   *                                                                       AIS     PROSEQ    594   A
PROSEQ   *        ENTRY  B2 - RIF, LOC OF R1 OF FIRST INS. IN SEQ.               AIS     PROSEQ    595   A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ   *                B3 - RIL, LOC OF R1 OF INS. MARKING END OF BLOCK.      AIS     PROSEQ       596     A
PROSEQ   *                A0 - O.SEQ                                             AIS     PROSEQ       597     A
PROSEQ   *                                                                       AIS     PROSEQ       598     A
PROSEQ   *         EXIT   A0 - PRESERVED.                                        AIS     PROSEQ       599     A
PROSEQ                                                                           AIS     PROSEQ       600     A
PROSEQ    BDUA     BFMW   D,(PRS,RJRS,PI),59-D.RFP                                         FCC0586        4     A
PROSEQ                                                                                     FCC0586        5     A
PROSEQ    BDU      ROUTINE                                                       AIS     PROSEQ       601     A
PROSEQ             SB2    B2-4        R1I = R1F - 4                              AIS     PROSEQ       602     A
PROSEQ             SA1    O.DUT                                                  AIS     PROSEQ       603     A
PROSEQ             SB7    X1          DUI = O.DUT                                AIS     PROSEQ       604     A
PROSEQ                                                                           AIS     PROSEQ       605     A
PROSEQ    BDU1     SB2    B2+4        R1I = R1I + 4                              AIS     PROSEQ       606     A
PROSEQ             MX0    -D.USESL                                               AIS     PROSEQ       607     A
PROSEQ             SA1    B2+2        DI = R1I + 2 ;  DIW = [DI]                 AIS     PROSEQ       608     A
PROSEQ             EQ     B2,B3,BDU   IF R1I = R1L                               AIS     PROSEQ       609     A
PROSEQ             SX5    B7          II = DUI                                   AIS     PROSEQ       610     A
PROSEQ             LX5    DP.IIP                                                 AIS     PROSEQ       611     A
PROSEQ             LX1    -D.USESP                                               AIS     PROSEQ       612     A
PROSEQ             SX6    B7          NI = DUI                                   AIS     PROSEQ       613     A
PROSEQ             BX2    -X0*X1      U = USES[DIW]                              AIS     PROSEQ       614           I
          -FCC0586
PROSEQ                                                                                     PROSEQ       615           I
          -FCC0586
PROSEQ   *         USES FIELD OF NON-DEFINITION INSTUCTIONS CONTAINS OPT=2 INFO.           PROSEQ       616           I
          -FCC0586
PROSEQ                                                                                     PROSEQ       617           I
          -FCC0586
PROSEQ             BX3    X1                                                                PROSEQ       618           I
          -FCC0586
PROSEQ             LX3    D.USESP+59-D.^DP                                                  PROSEQ       619           I
          -FCC0586
PROSEQ             AX3    59                                                                PROSEQ       620           I
          -FCC0586
PROSEQ             BX2    -X3*X2      IF ^D[DIW] THEN U=0                                   PROSEQ       621           I
          -FCC0586
PROSEQ             LX6    DP.NIP                                                 AIS     PROSEQ       622           I
          -FCC0586
PROSEQ             BX7    X6+X5                                                  AIS     PROSEQ       623           I
          -FCC0586
PROSEQ             SB7    B7+X2       DUI = DUI + U                              AIS     PROSEQ       624           I
          -FCC0586
PROSEQ             SA7    A1+B1       LI = DI + 1 ;  [LI] = DPW(II,NI)           AIS     PROSEQ       625           I
          -FCC0586
PROSEQ             LX1    D.USESP+59-D.RFP                                       AIS     PROSEQ       626           I
          -FCC0586
PROSEQ             SA2    B2          R1W = [R1I]                                AIS     PROSEQ       627           I
          -FCC0586
PROSEQ             UX0    B6,X2       OCI = OC[R1W]                              AIS     PROSEQ       628           I
          -FCC0586
PROSEQ             PL     X1,BDU2     IF ^RF[DIW]                                AIS     PROSEQ       629           I
          -FCC0586

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            SB5    B6-OC.RS                                          AIS     PROSEQ       630          I
          -FCC0586
PROSEQ            ZR     B5,BDU2     IF OCI = OC.RS                        AIS     PROSEQ       631          I
          -FCC0586
PROSEQ            SA2    B2          R1W = [R1I]                                   FCC0586        6     A
PROSEQ            LX6    DP.NIP                                                    FCC0586        7     A
PROSEQ            BX3    -X0*X1      U = USES[DIW]                                 FCC0586        8     A
PROSEQ            BX7    X6+X5                                                     FCC0586        9     A
PROSEQ            SA7    A1+B1       LI = DI + 1 ;  [LI] = DPW(II,NI)              FCC0586       10     A
PROSEQ            UX0    B6,X2       OCI = OC[R1W]                                 FCC0586       11     A
PROSEQ            LX1    D.USESP+59-D.^DP                                          FCC0586       12     A
PROSEQ            MI     X1,BDU2     IF ^D[DIW]                                    FCC0586       13     A
PROSEQ            BX0    X1                                                        FCC0586       14     A
PROSEQ            SB7    B7+X3       DUI = DUI + U                                 FCC0586       15     A
PROSEQ            LX0    D.^DP-D.RFP                                               FCC0586       16     A
PROSEQ            PL     X0,BDU2     IF ^RF[DIW]                                   FCC0586       17     A
PROSEQ            SA3    BDUA                                                      FCC0586       18     A
PROSEQ            BX0    X3*X0                                                     FCC0586       19     A
PROSEQ            NZ     X0,BDU2     IF (PRS,RJRS,PI)[DIW]                         FCC0586       20     A
PROSEQ            BX0    X1                                                        CCGA099       26     A
PROSEQ            LX0    D.^DP+1-D.XMTP                                            CCGA099       27          I
          -CCGA100
PROSEQ            LX0    D.^DP-D.XMTP                                              CCGA100       22     A
PROSEQ            MI     X0,BDU2     IF LEVEL 0 & LCM                              CCGA099       28     A
PROSEQ                                                                     AIS     PROSEQ       632     A
PROSEQ    *       FILE *RF* DEFINITION IN BCT.                             AIS     PROSEQ       633     A
PROSEQ                                                                     AIS     PROSEQ       634     A
PROSEQ            SA4    L.BCT                                             AIS     PROSEQ       635     A
PROSEQ            SA3    O.BCT                                             AIS     PROSEQ       636     A
PROSEQ            MX0    -R1.RIL                                           AIS     PROSEQ       637     A
PROSEQ            SB5    X4                                                        PROSEQ       638     A
PROSEQ            LX2    -R1.RIP                                           AIS     PROSEQ       639     A
PROSEQ            BX6    -X0*X2                                            AIS     PROSEQ       640     A
PROSEQ            LX6    BC.RIP      RID = RI[R1W]                         AIS     PROSEQ       641     A
PROSEQ            SA6    X3+B5       BCT(L.BCT) = BCW(RID)                         PROSEQ       642     A
PROSEQ            SX7    B5+B1                                                     PROSEQ       643     A
PROSEQ            LX2    R1.RIP                                            AIS     PROSEQ       644     A
PROSEQ            SA7    A4          L.BCT = L.BCT + 1                     AIS     PROSEQ       645     A
PROSEQ                                                                     AIS     PROSEQ       646     A
PROSEQ     BDU2   LX1    1+D.RFP+58-D.TYP                                  AIS     PROSEQ       647          I
          -FCC0586
PROSEQ     BDU2   LX1    1+D.^DP+58-D.TYP                                          FCC0586       21     A
PROSEQ            IX7    X1+X1                                             AIS     PROSEQ       648     A
PROSEQ            MI     X1,BDU6     IF TY[DIW] .GE. III                   AIS     PROSEQ       649     A
PROSEQ            MI     X7,BDU5     IF TY[DIW] = II                       AIS     PROSEQ       650     A
PROSEQ                                                                     AIS     PROSEQ       651     A
PROSEQ    *       TYPE I INSTRUCTION.                                      AIS     PROSEQ       652     A
PROSEQ                                                                     AIS     PROSEQ       653     A
PROSEQ            LX1    2+D.TYP-D.TYIOP                                   AIS     PROSEQ       654     A
PROSEQ            MX6    -D.TYIOL                                          AIS     PROSEQ       655     A
PROSEQ            SB5    B6-OC.PLD                                         AIS     PROSEQ       656     A
PROSEQ            BX7    -X6*X1                                            AIS     PROSEQ       657     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            MX0    -R1.RIL                                    AIS     PROSEQ       658     A
PROSEQ            SA3    =XTYIOM+X7                                 AIS     PROSEQ       659     A
PROSEQ            UX6    B4,X3                                      AIS     PROSEQ       660     A
PROSEQ            BX3    X6*X2                                      AIS     PROSEQ       661     A
PROSEQ            AX4    B4,X3                                      AIS     PROSEQ       662     A
PROSEQ            BX3    -X0*X4                                     AIS     PROSEQ       663     A
PROSEQ                                                              AIS     PROSEQ       664     A
PROSEQ    BDU3    ZR     X3,BDU4     IF OPN = 0                     AIS     PROSEQ       665     A
PROSEQ            RJ     EDU         ENTER USE IN DUT               AIS     PROSEQ       666     A
PROSEQ            MI     B5,BDU4     IF OCI .LT. OC.PLD             AIS     PROSEQ       667     A
PROSEQ            ZR     B5,BDU3A    IF OCI = OC.PLD                AIS     PROSEQ       668     A
PROSEQ            NE     B5,B1,BDU4  IF OCI .NE. OC.PST             AIS     PROSEQ       669     A
PROSEQ            ZR     B4,BDU4     IF RI MOD                      AIS     PROSEQ       670     A
PROSEQ                                                              AIS     PROSEQ       671     A
PROSEQ    *       SHORT USE. SET SU BIT IN DUT ENTRY.               AIS     PROSEQ       672     A
PROSEQ                                                              AIS     PROSEQ       673     A
PROSEQ    BDU3A   MX3    1                                          AIS     PROSEQ       674     A
PROSEQ            LX3    1+DU.SUP                                   AIS     PROSEQ       675     A
PROSEQ            BX7    X7+X3       SU[DUW] = 1                    AIS     PROSEQ       676     A
PROSEQ            SA7    A7                                         AIS     PROSEQ       677     A
PROSEQ                                                              AIS     PROSEQ       678     A
PROSEQ    BDU4    AX4    R1.RIL                                     AIS     PROSEQ       679     A
PROSEQ            ZR     X4,BDU1     IF NO OPERANDS REMAIN          AIS     PROSEQ       680     A
PROSEQ            SB4    B4+R1.RIL                                  AIS     PROSEQ       681     A
PROSEQ            BX3    -X0*X4      OPN =                          AIS     PROSEQ       682     A
PROSEQ            EQ     BDU3                                       AIS     PROSEQ       683     A
PROSEQ                                                              AIS     PROSEQ       684     A
PROSEQ    *       TYPE II INSTRUCTION.                              AIS     PROSEQ       685     A
PROSEQ                                                              AIS     PROSEQ       686     A
PROSEQ    BDU5    LX1    1+D.TYP-D.PIP                              AIS     PROSEQ       687     A
PROSEQ            PL     X1,BDU1     IF ^PI[DIW]                    AIS     PROSEQ       688     A
PROSEQ                                                              AIS     PROSEQ       689     A
PROSEQ    *       RS OR DEF INSTRUCTION.                            AIS     PROSEQ       690     A
PROSEQ                                                              AIS     PROSEQ       691     A
PROSEQ            BX3    X2                                                 PROSEQ       692     A
PROSEQ            LX1    D.PIP-D.^DP                                        PROSEQ       693     A
PROSEQ            PL     X1,BDU5A    IF D[DIW]                              PROSEQ       694     A
PROSEQ            LX3    59-R1.SOP-SO.LKP                                   PROSEQ       695     A
PROSEQ            MI     X3,BDU5A    IF MOD(LK[SO[R1W]],2) .NE. 0  */ ODD TYPE  PROSEQ   696     A
PROSEQ                                                                      PROSEQ       697     A
PROSEQ    *       ENTER RI OF TYPE 0 OR 2 RS IN DUT.                        PROSEQ       698     A
PROSEQ                                                                      PROSEQ       699     A
PROSEQ            MX0    -R1.RIL                                            PROSEQ       700     A
PROSEQ            R=     B4,R1.RIP                                          PROSEQ       701     A
PROSEQ            LX2    -R1.RIP                                            PROSEQ       702     A
PROSEQ            BX3    -X0*X2                                             PROSEQ       703     A
PROSEQ            RJ     EDU                                                PROSEQ       704     A
PROSEQ            SX6    B3-B2                                              FCC1837X       6     A
PROSEQ            LX6    DU.USEP                                            FCC1837X       7     A
PROSEQ            IX7    X6+X7       USE[DUT] = R1L-O.SEQ */ USED UNTIL NEXT BM  FCC1837X   8     A
PROSEQ            SA7    A7                                                 FCC1837X       9     A
PROSEQ            LX2    R1.RIP                                             PROSEQ       705     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ                                                                                  PROSEQ     706     A
PROSEQ    BDU5A    MX0    -SO.RTL                                                      PROSEQ     707     A
PROSEQ             LX2    -R1.SOP-SO.RTP                                       AIS     PROSEQ     708     A
PROSEQ             BX3    -X0*X2                                               AIS     PROSEQ     709     A
PROSEQ             SB6    X3-RT.BR                                             AIS     PROSEQ     710     A
PROSEQ             NZ     B6,BDU1     IF NOT B-REG LOCK                        AIS     PROSEQ     711     A
PROSEQ             LX2    SO.RTP-SO.RNP                                        AIS     PROSEQ     712     A
PROSEQ             BX3    -X0*X2                                               AIS     PROSEQ     713     A
PROSEQ             ZR     X3,BDU1     IF LOCK OF B0                            AIS     PROSEQ     714     A
PROSEQ             SB5    X3                                                   AIS     PROSEQ     715     A
PROSEQ                                                                         AIS     PROSEQ     716     A
PROSEQ    *        ENTER PRE-ASSIGNED B-REG IN PBT.                             AIS     PROSEQ     717     A
PROSEQ                                                                         AIS     PROSEQ     718     A
PROSEQ             SA5    L.PBT                                                AIS     PROSEQ     719     A
PROSEQ             SA4    O.PBT                                                AIS     PROSEQ     720     A
PROSEQ             LX2    SO.RNP+R1.SOP-R1.RIP                                 AIS     PROSEQ     721     A
PROSEQ             MX0    -R1.RIL                                              AIS     PROSEQ     722     A
PROSEQ             SB4    X5                                                   AIS     PROSEQ     723     A
PROSEQ             BX6    -X0*X2      RID = RI[R1W]                            AIS     PROSEQ     724     A
PROSEQ             LX6    PB.RIP                                               AIS     PROSEQ     725     A
PROSEQ             LX3    PB.RNP                                               AIS     PROSEQ     726     A
PROSEQ             BX6    X6+X3                                                AIS     PROSEQ     727     A
PROSEQ             LX2    R1.RIP+59-R1.SOP-SO.LKP-1                            AIS     PROSEQ     728     A
PROSEQ             IX3    X2+X2       LT = LK[SO[R1W]]                         AIS     PROSEQ     729     A
PROSEQ             SX5    LU.INF      LU = .INF.                               AIS     PROSEQ     730     A
PROSEQ             AX3    59                                                   AIS     PROSEQ     731     A
PROSEQ             BX1    -X3*X5      IF (LT .EQ. 1  .OR.  LT .EQ. 3) LU = 0   AIS     PROSEQ     732     A
PROSEQ             LX1    PB.LUP                                               AIS     PROSEQ     733     A
PROSEQ             BX6    X6+X1                                                AIS     PROSEQ     734     A
PROSEQ             SA6    X4+B4       PBT(L.PBT) = PBW(RN,PID,LU)              AIS     PROSEQ     735     A
PROSEQ             SX7    B4+B1                                                AIS     PROSEQ     736     A
PROSEQ             SA7    A5          L.PBT = L.PBT + 1                        AIS     PROSEQ     737     A
PROSEQ                                                                         AIS     PROSEQ     738     A
PROSEQ    *        SET FLAG (BIT IN FLF) TO TAKE REG OFF MARKET AT END OF       .NO.ID. PROSEQ     739     A
PROSEQ    *        SUBSEQUENCE IF THIS WAS A TYPE 2 LOCK.                       .NO.ID. PROSEQ     740     A
PROSEQ                                                                         AIS     PROSEQ     741     A
PROSEQ             MX0    1                                                    AIS     PROSEQ     742     A
PROSEQ             BX3    -X3*X2                                               AIS     PROSEQ     743     A
PROSEQ             BX2    X0*X3                                                AIS     PROSEQ     744     A
PROSEQ             SA4    FLF                                                  AIS     PROSEQ     745     A
PROSEQ             LX2    B5                                                   AIS     PROSEQ     746     A
PROSEQ             BX6    X4+X2                                                AIS     PROSEQ     747     A
PROSEQ             SA6    A4                                                   AIS     PROSEQ     748     A
PROSEQ             EQ     BDU1                                                 AIS     PROSEQ     749     A
PROSEQ                                                                         AIS     PROSEQ     750     A
PROSEQ    *        TYPE III OR IV INSTRUCTION.                                  AIS     PROSEQ     751     A
PROSEQ                                                                         AIS     PROSEQ     752     A
PROSEQ    BDU6     MI     X7,BDU1     IF TY[DIW] = IV                          AIS     PROSEQ     753     A
PROSEQ             LX1    1+D.TYP-D.^DP                                        AIS     PROSEQ     754     A
PROSEQ             SX0    RMSK                                                 AIS     PROSEQ     755     A
PROSEQ             PL     X1,BDU7     IF D[DIW]                                AIS     PROSEQ     756     A
PROSEQ             BX3    X0*X2                                                AIS     PROSEQ     757     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            ZR     X3,BDU7     IF RI[R1W] .LT. 2                          AIS     PROSEQ       758     A
PROSEQ            SB4    0           SC = 0                                     AIS     PROSEQ       759     A
PROSEQ            RJ     EDU         ENTER USE OF RI IN DUT                     AIS     PROSEQ       760     A
PROSEQ                                                                          AIS     PROSEQ       761     A
PROSEQ    BDU7    SA4    B2+B1       R2 = R1I + 1 ;  R2W = [R2]                 AIS     PROSEQ       762     A
PROSEQ            LX4    -IH.RFP                                                AIS     PROSEQ       763     A
PROSEQ            BX3    X0*X4                                                  AIS     PROSEQ       764     A
PROSEQ            ZR     X3,BDU1     IF RF[R2W] .LT. 2                          AIS     PROSEQ       765     A
PROSEQ            SB4    -IH.RFP     SC = IH.RFP                                AIS     PROSEQ       766     A
PROSEQ            RJ     EDU         FILE USE OF RF IN DUT                      AIS     PROSEQ       767     A
PROSEQ                                                                          AIS     PROSEQ       768     A
PROSEQ    *       MARK SHORT USE.                                               AIS     PROSEQ       769     A
PROSEQ                                                                          AIS     PROSEQ       770     A
PROSEQ            MX3    1                                                      AIS     PROSEQ       771     A
PROSEQ            LX3    1+DU.SUP                                               AIS     PROSEQ       772     A
PROSEQ            BX7    X7+X3       SU[DUW] = 1                                AIS     PROSEQ       773     A
PROSEQ            SA7    A7                                                     AIS     PROSEQ       774     A
PROSEQ            EQ     BDU1                                                   AIS     PROSEQ       775     A
PROSEQ    EDU     SPACE  3,14        EDU                                                PROSEQ       776     A
PROSEQ    **      EDU - ENTER USE IN DUT.                                       AIS     PROSEQ       777     A
PROSEQ    *       ENTRY  X3 - R-NUMBER OF USED OPERAND.                         .NO.ID. PROSEQ       778     A
PROSEQ    *                B2 - R1I, *SEQ* INDEX OF USE.                        .NO.ID. PROSEQ       779     A
PROSEQ    *                B4 - SC, SHIFT COUNT .                               AIS     PROSEQ       780     A
PROSEQ    *                A0 - O.SEQ                                           AIS     PROSEQ       781     A
PROSEQ    *                                                                     AIS     PROSEQ       782     A
PROSEQ    *       EXIT   A0 - PRESERVED.                                        AIS     PROSEQ       783         I
          -FCC1837X
PROSEQ    *       EXIT   A0, B4 PRESERVED                                               FCC1837X      10     A
PROSEQ    *                A7,X7 - ADDRESS AND CONTENTS OF UPDATED *DUT* WORD.          FCC1837X      11     A
PROSEQ    *                                                                     AIS     PROSEQ       784     A
PROSEQ    *       USES - X3,X5,X6,X7,A3,A5,A6,A7,B6                             AIS     PROSEQ       785     A
PROSEQ                                                                          AIS     PROSEQ       786     A
PROSEQ    EDU     ROUTINE                                                       AIS     PROSEQ       787     A
PROSEQ            SB6    A0                                                     AIS     PROSEQ       788     A
PROSEQ            SX6    B2-B6       USE = R1I - O.SEQ                          AIS     PROSEQ       789     A
PROSEQ            SB6    A0+3                                                   AIS     PROSEQ       790     A
PROSEQ            SA5    X3+B6       LD = O.SEQ + DEF + 3                       AIS     PROSEQ       791     A
PROSEQ            LX5    -DP.NIP                                                AIS     PROSEQ       792     A
PROSEQ            LX3    DU.DEFP                                                AIS     PROSEQ       793     A
PROSEQ            LX6    DU.USEP                                                AIS     PROSEQ       794     A
PROSEQ            BX7    X6+X3                                                  AIS     PROSEQ       795     A
PROSEQ            SX3    B1                                                     .NO.ID. PROSEQ       796     A
PROSEQ            PX7    B4,X7                                                  AIS     PROSEQ       797     A
PROSEQ            SA7    X5          DUT(NI[LD]) = DUW(0,SC,USE,DEF)            AIS     PROSEQ       798     A
PROSEQ            IX6    X5+X3       NI[LD] = NI[LD] + 1                        AIS     PROSEQ       799     A
PROSEQ            LX6    DP.NIP                                                 AIS     PROSEQ       800     A
PROSEQ            SA6    A5                                                     AIS     PROSEQ       801     A
PROSEQ            LX5    DP.NIP-DP.IIP                                                  FCC1837X      12     A
PROSEQ            SB6    X5                                                             FCC1837X      13     A
PROSEQ            SX3    A7-B6                                                          FCC1837X      14     A
PROSEQ            ZR     X3,EDU      IF II[LD] = NI[LD] - 1  */ FIRST USE               FCC1837X      15     A
PROSEQ                                                                                  FCC1837X      16     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ   *         ENSURE THAT THE LAST USE IS THE MAX.  REVERSAL CAN HAPPEN          FCC1837X     17    A
PROSEQ   *         WHEN USES > 1 AND AN EARLY USE *PRS* TO A UJP LOCK *RS*.           FCC1837X     18    A
PROSEQ                                                                                FCC1837X     19    A
PROSEQ             SA5    A7-B1                                                       FCC1837X     20    A
PROSEQ             LX7    -DU.USEP                                                    FCC1837X     21    A
PROSEQ             SX6    X7                                                          FCC1837X     22    A
PROSEQ             LX5    -DU.USEP                                                    FCC1837X     23    A
PROSEQ             SX3    X5                                                          FCC1837X     24    A
PROSEQ             IX3    X6-X3                                                       FCC1837X     25    A
PROSEQ             LX7    DU.USEP                                                     FCC1837X     26    A
PROSEQ             PL     X3,EDU      IF USE[NI] \ USE[NI-1]                          FCC1837X     27    A
PROSEQ             LX5    DU.USEP                                                     FCC1837X     28    A
PROSEQ             SA7    A7-B1       SWAP( DUT(NI) , DUT(NI-1) )                     FCC1837X     29    A
PROSEQ             BX7    X5                                                          FCC1837X     30    A
PROSEQ             SA7    A7+B1                                                       FCC1837X     31    A
PROSEQ             EQ     EDU                                                   AIS   PROSEQ      802    A
PROSEQ    IUP      EJECT              IUP                                        AIS   PROSEQ      803    A
PROSEQ   **        IUP - INSTALL USE INFORMATION IN PBT.                         AIS   PROSEQ      804    A
PROSEQ   *                                                                       AIS   PROSEQ      805    A
PROSEQ   *         ENTRY  A0 - O.SEQ                                             AIS   PROSEQ      806    A
PROSEQ   *                                                                       AIS   PROSEQ      807    A
PROSEQ   *         EXIT   A0 - PRESERVED.                                        AIS   PROSEQ      808    A
PROSEQ                                                                           AIS   PROSEQ      809    A
PROSEQ    IUP      ROUTINE                                                       AIS   PROSEQ      810    A
PROSEQ             SA1    L.PBT                                                 AIS   PROSEQ      811    A
PROSEQ             SA2    O.PBT                                                 AIS   PROSEQ      812    A
PROSEQ             SB7    X2                                                    AIS   PROSEQ      813    A
PROSEQ             SA1    X1+B7       PBC = PBT(L.PBT+1)                        AIS   PROSEQ      814    A
PROSEQ             SB7    A0+3                                                  AIS   PROSEQ      815    A
PROSEQ             MX0    -PB.RNL                                               AIS   PROSEQ      816    A
PROSEQ             EQ     IUP2                                                  AIS   PROSEQ      817    A
PROSEQ                                                                           AIS   PROSEQ      818    A
PROSEQ    IUP1     LX1    PB.LUP                                                AIS   PROSEQ      819    A
PROSEQ             LX2    PB.PP       P = PI                                    AIS   PROSEQ      820    A
PROSEQ             BX6    X1+X2       P[PBW] = P                                AIS   PROSEQ      821    A
PROSEQ             SA6    A1+                                                   AIS   PROSEQ      822    A
PROSEQ                                                                           AIS   PROSEQ      823    A
PROSEQ    IUP2     SA1    A1-B1       PBC = PBC - 1 ;  PBW = [PBC]              AIS   PROSEQ      824    A
PROSEQ             ZR     X1,IUP      IF PBC = PBT(0)                           AIS   PROSEQ      825    A
PROSEQ             LX1    -PB.RNP                                               AIS   PROSEQ      826    A
PROSEQ             BX3    -X0*X1                                                AIS   PROSEQ      827    A
PROSEQ             SA2    PBI+X3-1    PI = PBI(RN[PBW]) ;  PIW = [PI]           AIS   PROSEQ      828    A
PROSEQ             BX4    X1                                                    AIS   PROSEQ      829    A
PROSEQ             LX2    59-PI.NAP                                             AIS   PROSEQ      830    A
PROSEQ             MI     X2,IUP2     IF NA[PIW]                                AIS   PROSEQ      831    A
PROSEQ             LX2    1+PI.NAP+PB.PP            P = PI                      AIS   PROSEQ      832    A
PROSEQ             SX6    A1                                                    AIS   PROSEQ      833    A
PROSEQ             LX1    PB.RNP-PB.LUP                                         AIS   PROSEQ      834    A
PROSEQ             SA6    A2          P[PI] = PBC                               AIS   PROSEQ      835    A
PROSEQ             SX3    X1                                                    AIS   PROSEQ      836    A
PROSEQ             NZ     X3,IUP1     IF LU[PBW] .NE. 0                         AIS   PROSEQ      837    A
PROSEQ                                                                           AIS   PROSEQ      838    A

             0         1         2         3         4         5         6         7         8
             12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ   *         INSTALL LAST USE IN PBT ENTRY FOR TEMP LOCKS.                AIS     PROSEQ       839     A
PROSEQ                                                                          AIS     PROSEQ       840     A
PROSEQ             LX4    PB.RNP-PB.RIP                                         AIS     PROSEQ       841     A
PROSEQ             SA3    X4+B7       LI = O.SEQ + RI[PBW] + 3 ;  LIW = [LI]    AIS     PROSEQ       842     A
PROSEQ             LX3    -DP.NIP                                               AIS     PROSEQ       843     A
PROSEQ             SA4    X3-1        DUC = DUT(NI[LIW]-1) ;  DUW = [DUC]       AIS     PROSEQ       844     A
PROSEQ             LX4    -DU.USEP                                              AIS     PROSEQ       845     A
PROSEQ             SX3    X4                                                    AIS     PROSEQ       846     A
PROSEQ             BX1    X1+X3       LU[PBW] = USE[DUW]                        AIS     PROSEQ       847     A
PROSEQ             EQ     IUP1                                                  AIS     PROSEQ       848     A
PROSEQ    IXU      EJECT              IXU                                              PROSEQ       849     A
PROSEQ   **        IXU - INSTALL EXTENDED USE SPAN IN BCT MEMBERS.              AIS     PROSEQ       850     A
PROSEQ   *                                                                      AIS     PROSEQ       851     A
PROSEQ   *         ENTRY  A0 - O.SEQ                                            AIS     PROSEQ       852     A
PROSEQ   *                                                                      AIS     PROSEQ       853     A
PROSEQ   *         EXIT   X4 - SMAXW ,  MAX REGW OVER EXTENDED USE OF ALL       AIS     PROSEQ       854     A
PROSEQ   *                CANDIDATES.                                           AIS     PROSEQ       855     A
PROSEQ   *                A0 - PRESERVED.                                       AIS     PROSEQ       856     A
PROSEQ                                                                          AIS     PROSEQ       857     A
PROSEQ    IXU      ROUTINE                                                      AIS     PROSEQ       858     A
PROSEQ             SA1    O.BCT                                                 AIS     PROSEQ       859     A
PROSEQ             SA2    L.BCT                                                 AIS     PROSEQ       860     A
PROSEQ             MX6    0                                                     AIS     PROSEQ       861     A
PROSEQ             SB6    X2                                                    AIS     PROSEQ       862     A
PROSEQ             BX4    X4-X4       SMAXW = 0                                 AIS     PROSEQ       863     A
PROSEQ             SA6    X1+B6       BCT(L.BCT+1)=0                            AIS     PROSEQ       864     A
PROSEQ             SA1    X1-1        N = 0                                     AIS     PROSEQ       865     A
PROSEQ             SB6    A0+3        OLS = O.SEQ + 3                           AIS     PROSEQ       866     A
PROSEQ             SB5    A0+2        ODS = O.SEQ + 2                           AIS     PROSEQ       867     A
PROSEQ             EQ     IXU3                                                  AIS     PROSEQ       868     A
PROSEQ                                                                          AIS     PROSEQ       869     A
PROSEQ    IXU1     IX3    X4-X7                                                 AIS     PROSEQ       870     A
PROSEQ             PL     X3,IXU2     IF SMAXW \ MAXR                           AIS     PROSEQ       871     A
PROSEQ             BX4    X7          SMAXW = MAXR                              AIS     PROSEQ       872     A
PROSEQ                                                                          AIS     PROSEQ       873     A
PROSEQ    IXU2     LX1    BC.RIP                                                AIS     PROSEQ       874     A
PROSEQ             BX3    X6+X1                                                 AIS     PROSEQ       875     A
PROSEQ             LX7    BC.MRWP                                               AIS     PROSEQ       876     A
PROSEQ             BX6    X3+X7                                                 AIS     PROSEQ       877     A
PROSEQ             SA6    A1                                                    AIS     PROSEQ       878     A
PROSEQ                                                                          AIS     PROSEQ       879     A
PROSEQ    IXU3     SA1    A1+B1       N = N + 1 ; BCN = BCT(N) ;  BCW = [BCN]   AIS     PROSEQ       880     A
PROSEQ             ZR     X1,IXU      IF N .GT. L.BCT                           AIS     PROSEQ       881     A
PROSEQ             LX1    -BC.RIP                                               AIS     PROSEQ       882     A
PROSEQ             SA5    B6+X1       LN = OLS + RI[BCW] ;  LNW = [LN]          AIS     PROSEQ       883     A
PROSEQ             LX5    -DP.NIP                                               AIS     PROSEQ       884     A
PROSEQ             SB2    X5          UEN = NI[LNW]                             AIS     PROSEQ       885     A
PROSEQ             LX5    DP.NIP-DP.IIP                                         AIS     PROSEQ       886     A
PROSEQ             SB3    X5          UI = USN = II[LNW]                        AIS     PROSEQ       887     A
PROSEQ             MX6    0           LXU = 0                                   AIS     PROSEQ       888     A
PROSEQ                                                                          AIS     PROSEQ       889     A
PROSEQ    IXU4     SA2    B3          UIW = [UI]                                AIS     PROSEQ       890     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            LX2    -DU.USEP                                          AIS    PROSEQ     891    A
PROSEQ            SA5    B6+X2       LU = OLS + USE[UIW] ;  LUW = [LU]     AIS    PROSEQ     892    A
PROSEQ            SB3    B3+B1       UI = UI + 1                           AIS    PROSEQ     893    A
PROSEQ            LX5    -DP.NIP                                           AIS    PROSEQ     894    A
PROSEQ            SA2    X5-1        XU = DU(NI[LUW]-1) ;  XUW = [XU]      AIS    PROSEQ     895    A
PROSEQ            LX2    -DU.USEP    XUI = USE[XUW]                        AIS    PROSEQ     896    A
PROSEQ            SX5    X2                                                AIS    PROSEQ     897    A
PROSEQ            IX7    X6-X5                                             AIS    PROSEQ     898    A
PROSEQ            PL     X7,IXU5     IF XUI .LE. LXU                       AIS    PROSEQ     899    A
PROSEQ            BX6    X5          LXU = XUI                             AIS    PROSEQ     900    A
PROSEQ                                                                     AIS    PROSEQ     901    A
PROSEQ    IXU5    NE     B3,B2,IXU4  IF UI .NE. UEN                        AIS    PROSEQ     902    A
PROSEQ            SX7    B0          MAXR = 0                              AIS    PROSEQ     903    A
PROSEQ            MX0    -D.REGWL                                          AIS    PROSEQ     904    A
PROSEQ            SB3    X6+B5       DJE = ODS + LXU                       AIS    PROSEQ     905    A
PROSEQ            LX6    BC.LXUP                                           AIS    PROSEQ     906    A
PROSEQ            SB2    X1+B5       DJI = DJS = ODS + RI[BCW]             AIS    PROSEQ     907    A
PROSEQ                                                                     AIS    PROSEQ     908    A
PROSEQ    IXU6    GT     B2,B3,IXU1  IF DJI .GT. DJE                       AIS    PROSEQ     909    A
PROSEQ            SA2    B2          DJW = [DJI]                           AIS    PROSEQ     910    A
PROSEQ            LX2    -D.REGWP                                          AIS    PROSEQ     911    A
PROSEQ            BX3    -X0*X2      RW = REGW[DJW]                        AIS    PROSEQ     912    A
PROSEQ            IX5    X7-X3                                             AIS    PROSEQ     913    A
PROSEQ            SB2    B2+4        DJI = DJI + 4                         AIS    PROSEQ     914    A
PROSEQ            PL     X5,IXU6     IF MAXR .GE. RW                       AIS    PROSEQ     915    A
PROSEQ            BX7    X3          MAXR = RW                             AIS    PROSEQ     916    A
PROSEQ            EQ     IXU6                                              AIS    PROSEQ     917    A
PROSEQ    ABR     EJECT              ABR                                          PROSEQ     918    A
PROSEQ    **      ABR - ASSIGN B REGISTERS.                                AIS    PROSEQ     919    A
PROSEQ    *       SCAN BCT, ASSIGNING AS MANY CANDIDATES AS POSSIBLE.      AIS    PROSEQ     920    A
PROSEQ    *                                                                AIS    PROSEQ     921    A
PROSEQ    *       ENTRY  A0 - O.SEQ                                        AIS    PROSEQ     922    A
PROSEQ    *                                                                AIS    PROSEQ     923    A
PROSEQ    *       EXIT   A0 - PRESERVED.                                   AIS    PROSEQ     924    A
PROSEQ                                                                     AIS    PROSEQ     925    A
PROSEQ    ABR     ROUTINE                                                  AIS    PROSEQ     926    A
PROSEQ            SETZERO  CAT,7                                           AIS    PROSEQ     927    A
PROSEQ            SA1    O.BCT                                             AIS    PROSEQ     928    A
PROSEQ            SA2    L.BCT                                             AIS    PROSEQ     929    A
PROSEQ            SB6    X1                                                AIS    PROSEQ     930    A
PROSEQ            SA6    X2+B6       BCT(L.BCT+1) = 0                      AIS    PROSEQ     931    A
PROSEQ            SA1    B6-B1       BI = O.BCT - 1                        AIS    PROSEQ     932    A
PROSEQ                                                                     AIS    PROSEQ     933    A
PROSEQ    ABR1    SA1    A1+B1       BI = BI + 1 ;  BIW = [BI]             AIS    PROSEQ     934    A
PROSEQ            ZR     X1,ABR      IF BIW = 0    */ END OF BCT           AIS    PROSEQ     935    A
PROSEQ                                                                           PROSEQ     936    A
PROSEQ    *       AVOID ASSIGNING CANDIDATE WITH NO SHORT USES.                  PROSEQ     937    A
PROSEQ                                                                           PROSEQ     938    A
PROSEQ            SB5    A0+3                                                    PROSEQ     939    A
PROSEQ            BX6    X1                                                      PROSEQ     940    A
PROSEQ            LX6    -BC.RIP                                                 PROSEQ     941    A
PROSEQ            SA2    X6+B5                                                   PROSEQ     942    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            LX2    -DP.NIP                                          PROSEQ     943    A
PROSEQ            SB4    X2          LP = DUT(NI[RI[BIW]+3])              PROSEQ     944    A
PROSEQ            LX2    DP.NIP-DP.IIP                                    PROSEQ     945    A
PROSEQ            SB3    X2          IP = DUT(II[RI[BIW]+3])              PROSEQ     946    A
PROSEQ                                                                    PROSEQ     947    A
PROSEQ    ABR1A   EQ     B3,B4,ABR1  IF IP = LP                           PROSEQ     948    A
PROSEQ            SA2    B3          IPW = [IP]                           PROSEQ     949    A
PROSEQ            LX2    59-DU.SUP                                        PROSEQ     950    A
PROSEQ            SB3    B3+B1       IP = IP + 1                          PROSEQ     951    A
PROSEQ            PL     X2,ABR1A    IF ^SU[IPW]                          PROSEQ     952    A
PROSEQ            LX1    -BC.MRWP                                   AIS   PROSEQ     953    A
PROSEQ            SX0    X1-TRW                                     AIS   PROSEQ     954    A
PROSEQ            MI     X0,ABR1     IF MRW[BIW] .LT. TRW           AIS   PROSEQ     955    A
PROSEQ                                                                    FCC0508      4    A
PROSEQ    *       DONT ASSIGN CANDIDATE WHOSE FINAL USE IS 60-BIT.        FCC0508      5    A
PROSEQ                                                                    FCC0508      6    A
PROSEQ            SA2    B4-B1       FP = LP - 1 ;  FPW = [FP]            FCC0508      7    A
PROSEQ            LX2    59-DU.SUP                                        FCC0508      8    A
PROSEQ            PL     X2,ABR1     IF ^SU[FPW]                          FCC0508      9    A
PROSEQ            LX1    BC.MRWP-BC.RIP                             AIS   PROSEQ     956    A
PROSEQ            SB4    X1                                         AIS   PROSEQ     957    A
PROSEQ            SB3    -B4         -RIN  = -RI[BIW]               AIS   PROSEQ     958    A
PROSEQ            SB6    B0          RN = 0                         AIS   PROSEQ     959    A
PROSEQ            SB7    7                                          AIS   PROSEQ     960    A
PROSEQ                                                              AIS   PROSEQ     961    A
PROSEQ    *       SCAN FOR AVAILABLE B REGISTER WITH NO LOCKS IN FUTURE.  AIS   PROSEQ     962    A
PROSEQ                                                              AIS   PROSEQ     963    A
PROSEQ    ABR2    SA2    PBI+B6      PN = PBI(RN+1) ;  PNW = [PN]   AIS   PROSEQ     964    A
PROSEQ            EQ     B6,B7,ABR3  IF RN = 7                            PROSEQ     965    A
PROSEQ            LX2    59-PI.NAP                                  AIS   PROSEQ     966    A
PROSEQ            SB6    B6+B1       RN = RN +1                     AIS   PROSEQ     967    A
PROSEQ            MI     X2,ABR2     IF NA[PNW]    */  RN OFF MARKET AIS   PROSEQ     968    A
PROSEQ            NZ     X2,ABR2     IF P[PNW] .NE. 0  */ LOCK IN FUTURE AIS   PROSEQ     969    A
PROSEQ            SA4    CAT+B6-1    CN = CAT(RN) ;  CNW = [CN]     AIS   PROSEQ     970    A
PROSEQ            LX4    -CA.LXUP                                   AIS   PROSEQ     971    A
PROSEQ            SX7    X4+B3                                      AIS   PROSEQ     972    A
PROSEQ            MI     X7,ABR7     IF LXU[CNW] .LT. RI[BIW]       AIS   PROSEQ     973    A
PROSEQ            NE     B6,B7,ABR2  IF RN .NE. 7                   AIS   PROSEQ     974    A
PROSEQ                                                              AIS   PROSEQ     975    A
PROSEQ    *       LOOK FOR AVAILABLE B REGISTER WITH A *HOLE* IN FUTURE LOCKS  AIS   PROSEQ     976    A
PROSEQ    *       WHICH WILL ACCOMODATE THIS CANDIDATE/S EXTENDED-USE SPAN.    AIS   PROSEQ     977    A
PROSEQ                                                              AIS   PROSEQ     978    A
PROSEQ    ABR3    SB6    B0          RN = 0                         PROSEQ     979    A
PROSEQ                                                              AIS   PROSEQ     980    A
PROSEQ    ABR4    SA2    PBI+B6      PN = PBI(RN+1) ;  PNW = [PN]   AIS   PROSEQ     981    A
PROSEQ            EQ     B6,B7,ABR1  IF RN = 7    */ CANT ASSIGN THIS CANDIDATE  AIS   PROSEQ     982    A
PROSEQ            SB6    B6+B1       RN = RN + 1                    AIS   PROSEQ     983    A
PROSEQ            LX2    59-PI.NAP                                  AIS   PROSEQ     984    A
PROSEQ            MI     X2,ABR4     IF NA[PNW]    THIS REG OFF MARKET AIS   PROSEQ     985    A
PROSEQ            SA4    CAT+B6-1    CN = CAT[RN] ;  CNW = [CN]     AIS   PROSEQ     986    A
PROSEQ            LX4    -CA.LXUP                                   AIS   PROSEQ     987    A
PROSEQ            SX7    X4+B3                                      AIS   PROSEQ     988    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            PL     X7,ABR4     IF LXU[CNW] .GE. RIN                    AIS     PROSEQ      989     A
PROSEQ            LX2    1+PI.NAP-PI.PP  PL=P[PNW]                           AIS     PROSEQ      990     A
PROSEQ                                                                       AIS     PROSEQ      991     A
PROSEQ    ABR5    SA3    X2          PBJ = PL ;  PBW = [PBJ]                 AIS     PROSEQ      992     A
PROSEQ            LX3    -PB.PP                                              AIS     PROSEQ      993     A
PROSEQ            SX2    X3          PL = P[PBW]                             AIS     PROSEQ      994     A
PROSEQ            LX3    PB.PP-PB.LUP                                        AIS     PROSEQ      995     A
PROSEQ            SX7    X3+B3                                               AIS     PROSEQ      996     A
PROSEQ            PL     X7,ABR6     IF LU[PBW] .GT. RIN                     AIS     PROSEQ      997     A
PROSEQ            BX6    X2                                                          PROSEQ      998     A
PROSEQ            SA6    A2          P[PNW] = PL                                 PROSEQ      999     A
PROSEQ            NZ     X2,ABR5     IF PL .NE. 0                            AIS     PROSEQ     1000     A
PROSEQ                                                                       AIS     PROSEQ     1001     A
PROSEQ    *       NO LOCKS TO THIS REG PAST THIS CANDIDATE.                  AIS     PROSEQ     1002     A
PROSEQ                                                                       AIS     PROSEQ     1003     A
PROSEQ            EQ     ABR7                                                AIS     PROSEQ     1004     A
PROSEQ                                                                       AIS     PROSEQ     1005     A
PROSEQ    *       DETERMINE IF NEXT LOCK IS FAR AHEAD ENOUGH TO LEAVE *HOLE* AIS     PROSEQ     1006     A
PROSEQ    *       FOR THIS CANDIDATE/S SPAN.                                 AIS     PROSEQ     1007     A
PROSEQ                                                                       AIS     PROSEQ     1008     A
PROSEQ    ABR6    SA1    A1                                                          PROSEQ     1009     A
PROSEQ            LX1    -BC.LXUP                                                    PROSEQ     1010     A
PROSEQ            SX4    X1                                                  AIS     PROSEQ     1011     A
PROSEQ            LX3    PB.LUP-PB.RIP                                       AIS     PROSEQ     1012     A
PROSEQ            SX5    X3                                                  AIS     PROSEQ     1013     A
PROSEQ            IX7    X5-X4                                               AIS     PROSEQ     1014     A
PROSEQ            MI     X7,ABR4     IF LXU[BIW] .GT. RI[PBW]  */ OVERLAP    AIS     PROSEQ     1015     A
PROSEQ                                                                       AIS     PROSEQ     1016     A
PROSEQ    *       ASSIGNMENT MADE.  GENERATE MODS TO ISSUE SA-RS AND MODIFY  AIS     PROSEQ     1017     A
PROSEQ    *       INSTRUCTIONS WHICH USE CANDIDATE IN 60-BIT CAPACITY.       AIS     PROSEQ     1018     A
PROSEQ                                                                       AIS     PROSEQ     1019     A
PROSEQ    ABR7    SA2    A1          BIW = [BI]                              AIS     PROSEQ     1020     A
PROSEQ            BX6    X2                                                  AIS     PROSEQ     1021     A
PROSEQ            SA6    CAT+B6-1    CAT(RN) = BIW                           AIS     PROSEQ     1022     A
PROSEQ            RJ     ISA         ISSUE SA-RS TO MOD                      AIS     PROSEQ     1023     A
PROSEQ            RJ     IUM         ISSUE USE MODIFICATIONS                 AIS     PROSEQ     1024     A
PROSEQ                                                                       AIS     PROSEQ     1025     A
PROSEQ    *       DECREMENT MRW OF ALL CANDIDATES FOLLOWING WHICH ARE FULLY  AIS     PROSEQ     1026     A
PROSEQ    *       CONTAINED IN THE SPAN OF THE CANDIDATE JUST ASSIGNED.      AIS     PROSEQ     1027     A
PROSEQ                                                                       AIS     PROSEQ     1028     A
PROSEQ            SA2    A1          BJ = BI                                 AIS     PROSEQ     1029     A
PROSEQ            LX2    -BC.LXUP                                            AIS     PROSEQ     1030     A
PROSEQ            SB7    X2                                                  AIS     PROSEQ     1031     A
PROSEQ            SB5    -B7                                                 AIS     PROSEQ     1032     A
PROSEQ            SX1    B1                                                  AIS     PROSEQ     1033     A
PROSEQ                                                                       AIS     PROSEQ     1034     A
PROSEQ    ABR8    SA2    A2+B1       BJ = BJ + 1 ;  BJW = [BJ]               AIS     PROSEQ     1035     A
PROSEQ            ZR     X2,ABR1     IF BJW = 0    */ END OF BCT             AIS     PROSEQ     1036     A
PROSEQ            LX2    -BC.LXUP                                            AIS     PROSEQ     1037     A
PROSEQ            SX7    X2+B5                                               AIS     PROSEQ     1038     A
PROSEQ            PL     X7,ABR8     IF LXU[BJW] .GE. LXU[BIW]  */ OVERLAP   AIS     PROSEQ     1039     A
PROSEQ            LX2    BC.LXUP-BC.MRWP                                     AIS     PROSEQ     1040     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            SX7    X2                                          AIS    PROSEQ    1041    A
PROSEQ            ZR     X7,ABR8     IF MRW[BJW] = 0                 AIS    PROSEQ    1042    A
PROSEQ            IX6    X2-X1                                       AIS    PROSEQ    1043    A
PROSEQ            LX6    BC.MRWP                                     AIS    PROSEQ    1044    A
PROSEQ            SA6    A2          MRW[BJW] = MRW[BJW] - 1         AIS    PROSEQ    1045    A
PROSEQ            EQ     ABR8                                        AIS    PROSEQ    1046    A
PROSEQ    ISA     EJECT              ISA                                    PROSEQ    1047    A
PROSEQ    **      ISA - ISSUE SHORT ADD AND REGISTER STORE.          AIS    PROSEQ    1048    A
PROSEQ    *                                                          AIS    PROSEQ    1049    A
PROSEQ    *       ENTRY  A1 - BI, LOCATION OF BCT ENTRY FOR THIS ASSIGNMENT.  AIS    PROSEQ    1050    A
PROSEQ    *              B3 - (-RIN)                                 AIS    PROSEQ    1051    A
PROSEQ    *              B6 - RN, REGISTER TO BE ASSIGNED.           AIS    PROSEQ    1052    A
PROSEQ    *                                                          AIS    PROSEQ    1053    A
PROSEQ    *       EXIT   B3 - PRESERVED.                             AIS    PROSEQ    1054    A
PROSEQ    *              B4 - NMA, NEXT ADDRESS IN MOD.              AIS    PROSEQ    1055    A
PROSEQ    *              B7 - RIS, INDEX IN MOD OF SA GENERATED.     AIS    PROSEQ    1056    A
PROSEQ    *              A0 - PRESERVED.                             AIS    PROSEQ    1057    A
PROSEQ    *              A1 - PRESERVED.                             AIS    PROSEQ    1058    A
PROSEQ                                                               AIS    PROSEQ    1059    A
PROSEQ    ISA     ROUTINE                                            AIS    PROSEQ    1060    A
PROSEQ            SA2    O.MLT                                       AIS    PROSEQ    1061    A
PROSEQ            SA3    L.MLT                                       AIS    PROSEQ    1062    A
PROSEQ            SA5    O.MOD                                       AIS    PROSEQ    1063    A
PROSEQ            MX6    0                                           AIS    PROSEQ    1064    A
PROSEQ            IX7    X2+X3                                       AIS    PROSEQ    1065    A
PROSEQ            SB2    X5                                          AIS    PROSEQ    1066    A
PROSEQ            SA6    X7          MLT(L.MLT+1) = 0                AIS    PROSEQ    1067    A
PROSEQ            SA4    X2          MI = O.MLT ;  MIN = [MI]        AIS    PROSEQ    1068    A
PROSEQ            SA5    L.MOD                                       AIS    PROSEQ    1069    A
PROSEQ                                                               AIS    PROSEQ    1070    A
PROSEQ    ISA1    ZR     X4,ISA2     IF MIN = 0    */ END OF MLT     AIS    PROSEQ    1071    A
PROSEQ            LX4    -ML.IIP                                     AIS    PROSEQ    1072    A
PROSEQ            SX7    X4+B3       MIW = MIN                       AIS    PROSEQ    1073    A
PROSEQ            LX4    ML.IIP-ML.MTIP                              AIS    PROSEQ    1074    A
PROSEQ            SB5    X4                                          AIS    PROSEQ    1075    A
PROSEQ            SA4    A4+B1       MI = MI + 1 ;  MIN = [MI]       AIS    PROSEQ    1076    A
PROSEQ            NZ     X7,ISA1     IF II[MIW] .NE. RIN             AIS    PROSEQ    1077    A
PROSEQ            MX0    -R1.RIL                                     AIS    PROSEQ    1078    A
PROSEQ            SA2    B2+B5       TI = O.MOD+MTI[MIW]             AIS    PROSEQ    1079    A
PROSEQ            BX4    X0*X2                                       AIS    PROSEQ    1080    A
PROSEQ            SA3    A2+B1                                       AIS    PROSEQ    1081    A
PROSEQ            BX6    X4+X5       RI[TI] = L.MOD                  AIS    PROSEQ    1082    A
PROSEQ            LX7    X3                                          AIS    PROSEQ    1083    A
PROSEQ            SA6    X5+B2       [MOD(L.MOD+1)] = [TI]           AIS    PROSEQ    1084    A
PROSEQ            SA7    A6+B1       [MOD(L.MOD+2)] = [TI+1]         AIS    PROSEQ    1085    A
PROSEQ            SA2    A3+B1                                       AIS    PROSEQ    1086    A
PROSEQ            SA3    A2+B1                                       AIS    PROSEQ    1087    A
PROSEQ            BX6    X2                                          AIS    PROSEQ    1088    A
PROSEQ            LX7    X3                                          AIS    PROSEQ    1089    A
PROSEQ            SA6    A7+B1       [MOD(L.MOD+3)] = [TI+2]         AIS    PROSEQ    1090    A
PROSEQ            SA7    A6+B1       [MOD(L.MOD+4)] = [TI+3]         AIS    PROSEQ    1091    A
PROSEQ            SA4    A4-B1       MI = MI - 1                     AIS    PROSEQ    1092    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            EQ     ISA4                                          AIS     PROSEQ      1093     A
PROSEQ                                                                 AIS     PROSEQ      1094     A
PROSEQ    ISA2    SA2    L.MLT                                         AIS     PROSEQ      1095     A
PROSEQ            SX6    X2+B1                                         AIS     PROSEQ      1096     A
PROSEQ            SA6    A2          L.MLT = L.MLT + 1                 AIS     PROSEQ      1097     A
PROSEQ            MX0    -R1.RIL                                       AIS     PROSEQ      1098     A
PROSEQ            SA2    A0-B3       R1I = O.SEQ + RIN                 AIS     PROSEQ      1099     A
PROSEQ            SA3    A2+2        DI = R1I + 2                      AIS     PROSEQ      1100     A
PROSEQ            BX7    X0*X2                                         AIS     PROSEQ      1101     A
PROSEQ            SA2    A3-B1       R2I = DI - 1                      AIS     PROSEQ      1102     A
PROSEQ            BX6    X7+X5       RI = L.MOD                        AIS     PROSEQ      1103     A
PROSEQ            SX0    100000B                                       AIS     PROSEQ      1104     A
PROSEQ            LX3    58-D.TYP                                      AIS     PROSEQ      1105     A
PROSEQ            IX4    X3+X3                                         AIS     PROSEQ      1106     A
PROSEQ            LX0    IH.RFP                                        AIS     PROSEQ      1107     A
PROSEQ            IX7    X2+X0       RF = RF[R2I] + 100000             AIS     PROSEQ      1108     A
PROSEQ            BX4    X3+X4                                         AIS     PROSEQ      1109     A
PROSEQ            MI     X4,ISA3 IF TY[DI] .GT. 1                      AIS     PROSEQ      1110     A
PROSEQ            LX0    R1.RKP-IH.RFP                                 AIS     PROSEQ      1111     A
PROSEQ            IX4    X6+X0                                         AIS     PROSEQ      1112     A
PROSEQ            LX0    R1.RJP-R1.RKP                                 AIS     PROSEQ      1113     A
PROSEQ            IX6    X4+X0                                         AIS     PROSEQ      1114     A
PROSEQ                                                                 AIS     PROSEQ      1115     A
PROSEQ    ISA3    SA6    X5+B2                                         AIS     PROSEQ      1116     A
PROSEQ            SA7    A6+B1                                         AIS     PROSEQ      1117     A
PROSEQ            LX3    2+D.TYP                                       AIS     PROSEQ      1118     A
PROSEQ            BX6    X3                                            AIS     PROSEQ      1119     A
PROSEQ            MX7    0                                             AIS     PROSEQ      1120     A
PROSEQ            SA6    A7+B1                                         AIS     PROSEQ      1121     A
PROSEQ            SA7    A6+B1                                         AIS     PROSEQ      1122     A
PROSEQ                                                                 AIS     PROSEQ      1123     A
PROSEQ    ISA4    MX1    1                                             AIS     PROSEQ      1124     A
PROSEQ            LX1    1+ML.DELP   DELM = 1                          AIS     PROSEQ      1125     A
PROSEQ            MX3    2                                             AIS     PROSEQ      1126     A
PROSEQ            LX3    2+ML.NIP    NIM = 3                           AIS     PROSEQ      1127     A
PROSEQ            SX4    -B3                                           AIS     PROSEQ      1128     A
PROSEQ            LX4    ML.IIP      IIM = RIN                         AIS     PROSEQ      1129     A
PROSEQ            BX6    X1+X3                                         AIS     PROSEQ      1130     A
PROSEQ            LX5    ML.MTIP     MTIM = L.MOD                      AIS     PROSEQ      1131     A
PROSEQ            BX6    X4+X6                                         AIS     PROSEQ      1132     A
PROSEQ            BX7    X6+X5                                         AIS     PROSEQ      1133     A
PROSEQ            LX5    -ML.MTIP                                      AIS     PROSEQ      1134     A
PROSEQ            SA7    A4          [MI] = MLW(0,NIM,IIM,MTIM)        AIS     PROSEQ      1135     A
PROSEQ                                                                 AIS     PROSEQ      1136     A
PROSEQ            SB7    X5+4        LMN = L.MOD + 4                   AIS     PROSEQ      1137     A
PROSEQ            SB5    OC.SA                                         AIS     PROSEQ      1138     A
PROSEQ            SX6    B7          RIIN = LMN                        AIS     PROSEQ      1139     A
PROSEQ            LX6    R1.RIP                                        AIS     PROSEQ      1140     A
PROSEQ            LX5    R1.RJP      RJN = L.MOD                       AIS     PROSEQ      1141     A
PROSEQ            BX6    X6+X5                                         AIS     PROSEQ      1142     A
PROSEQ            MX7    0                                             AIS     PROSEQ      1143     A
PROSEQ            SB4    B7+B2                                         AIS     PROSEQ      1144     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            RJ     SRI                                                AIS     PROSEQ      1145     A
PROSEQ            SX3    SO.BLOCK                                           AIS     PROSEQ      1146     A
PROSEQ            SX1    B6                                                 AIS     PROSEQ      1147     A
PROSEQ            LX1    SO.RNP                                             AIS     PROSEQ      1148     A
PROSEQ            BX2    X1+X3                                              AIS     PROSEQ      1149     A
PROSEQ            LX2    R1.SOP                                             AIS     PROSEQ      1150     A
PROSEQ            SX7    B7                                                 AIS     PROSEQ      1151     A
PROSEQ            LX7    R1.RIP                                             AIS     PROSEQ      1152     A
PROSEQ            SB5    OC.RS                                              AIS     PROSEQ      1153     A
PROSEQ            BX6    X7+X2                                              AIS     PROSEQ      1154     A
PROSEQ            MX7    0                                                  AIS     PROSEQ      1155     A
PROSEQ            RJ     SRI                                                AIS     PROSEQ      1156     A
PROSEQ            SX7    B7+8                                               AIS     PROSEQ      1157     A
PROSEQ            SA7    L.MOD       L.MOD = LMN + 8                        AIS     PROSEQ      1158     A
PROSEQ            EQ     ISA                                                AIS     PROSEQ      1159     A
PROSEQ    IUM     EJECT              IUM                                            PROSEQ      1160     A
PROSEQ    **      IUM - ISSUE USE MODIFICATIONS.                            AIS     PROSEQ      1161     A
PROSEQ    *                                                                 AIS     PROSEQ      1162     A
PROSEQ    *       ENTRY  B3 - (-RIN), MINUS INDEX IN SEQ OF CANDIDATE DEFINITION AIS  PROSEQ    1163     A
PROSEQ    *              B4 - NEXT *MOD* STORE ADDRESS.                     AIS     PROSEQ      1164     A
PROSEQ    *              B7 - RIS, MOD INDEX OF SA.                         AIS     PROSEQ      1165     A
PROSEQ    *              A0 - O.SEQ                                         AIS     PROSEQ      1166     A
PROSEQ    *              A1 - BI, BCT ENTRY ADDRESS CORRESPONDING TO CANDIDATE.  AIS  PROSEQ    1167     A
PROSEQ    *                                                                 AIS     PROSEQ      1168     A
PROSEQ    *       EXIT   A0,A1 - PRESERVED.                                 AIS     PROSEQ      1169     A
PROSEQ                                                                      AIS     PROSEQ      1170     A
PROSEQ    IUMA    VFD    12/0,16/100000B,16/100000B,16/0                    AIS     PROSEQ      1171     A
PROSEQ                                                                      AIS     PROSEQ      1172     A
PROSEQ    IUM0    SA2    IUMB                                               AIS     PROSEQ      1173     A
PROSEQ            BX6    X2                                                 AIS     PROSEQ      1174     A
PROSEQ            SA6    A3          [UL] = SAVE                            AIS     PROSEQ      1175     A
PROSEQ                                                                      AIS     PROSEQ      1176     A
PROSEQ    IUM     ROUTINE                                                    AIS     PROSEQ      1177     A
PROSEQ            SB6    A0+3        OLS = O.SEQ + 3                        AIS     PROSEQ      1178     A
PROSEQ            SA3    B6-B3       LI = OLS + RIN ;  LIW = [LI]           AIS     PROSEQ      1179     A
PROSEQ            MX6    0                                                  AIS     PROSEQ      1180     A
PROSEQ            LX3    -DP.NIP                                            AIS     PROSEQ      1181     A
PROSEQ            SA4    X3          UL = NI[LIW]                           AIS     PROSEQ      1182     A
PROSEQ            BX7    X4          SAVE = [UL]                            AIS     PROSEQ      1183     A
PROSEQ            SA6    A4          [UL] = 0                               AIS     PROSEQ      1184     A
PROSEQ            SA7    IUMB                                               AIS     PROSEQ      1185     A
PROSEQ            LX3    DP.NIP-DP.IIP                                      AIS     PROSEQ      1186     A
PROSEQ            SB2    X3          UI = US = II[LIW]                      AIS     PROSEQ      1187     A
PROSEQ                                                                      AIS     PROSEQ      1188     A
PROSEQ    *       MODIFY ALL 60-BIT USES OF CANDIDATE SO THAT THESE USES    AIS     PROSEQ      1189     A
PROSEQ    *       ARE NOT CHANGED TO B-REG RI BY MPB.                       AIS     PROSEQ      1190     A
PROSEQ                                                                      AIS     PROSEQ      1191     A
PROSEQ    IUM1    SA3    B2          UIW = [UI]                             AIS     PROSEQ      1192     A
PROSEQ            ZR     X3,IUM0     IF UIW = 0    */ UI = UL               AIS     PROSEQ      1193     A
PROSEQ            SB2    B2+B1       UI = UI + 1                            AIS     PROSEQ      1194     A
PROSEQ            LX3    59-DU.SUP                                          AIS     PROSEQ      1195     A
PROSEQ            MI     X3,IUM1     IF SU[UIW]                             AIS     PROSEQ      1196     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            LX3    1+DU.SUP                                          AIS      PROSEQ      1197     A
PROSEQ            BX1    X3                                                AIS      PROSEQ      1198     A
PROSEQ            LX3    -DU.USEP                                          AIS      PROSEQ      1199     A
PROSEQ            SB6    A0                                                AIS      PROSEQ      1200     A
PROSEQ            SA2    B6+X3       R1I = O.SEQ + USE[UIW] ;  R1W = [R1I] AIS      PROSEQ      1201     A
PROSEQ            SA4    O.MLT                                             AIS      PROSEQ      1202     A
PROSEQ            SA5    L.MLT                                             AIS      PROSEQ      1203     A
PROSEQ            MX6    0                                                 AIS      PROSEQ      1204     A
PROSEQ            IX7    X4+X5                                             AIS      PROSEQ      1205     A
PROSEQ            MX0    -R1.RIL                                           AIS      PROSEQ      1206     A
PROSEQ            SA6    X7          MLT(L.MLT+1) = 0                      AIS      PROSEQ      1207     A
PROSEQ            SA4    X4-1        MI = O.MLT - 1                        AIS      PROSEQ      1208     A
PROSEQ            SB6    X3                                                AIS      PROSEQ      1209     A
PROSEQ            SB5    -B6         MU = -USE[UIW]                        AIS      PROSEQ      1210     A
PROSEQ            SX6    B7-4        RIM = RIS - 4                         AIS      PROSEQ      1211     A
PROSEQ                                                                     AIS      PROSEQ      1212     A
PROSEQ   *        SEARCH FOR *MOD* TO SAME INSTRUCTION.  IF WE FIND ONE,   AIS      PROSEQ      1213     A
PROSEQ   *        WE MODIFY THE FIRST INSTRUCTION IN IT.                   AIS      PROSEQ      1214     A
PROSEQ                                                                     AIS      PROSEQ      1215     A
PROSEQ    IUM2    SA4    A4+B1       MI = MI + 1 ;  MIN = [MI]             AIS      PROSEQ      1216     A
PROSEQ            ZR     X4,IUM3     IF MIN = 0    */ NO *MOD* ENTRY MATCHES AIS      PROSEQ      1217     A
PROSEQ            LX4    -ML.IIP                                           AIS      PROSEQ      1218     A
PROSEQ            SX7    X4+B5                                             AIS      PROSEQ      1219     A
PROSEQ            NZ     X7,IUM2     IF II[MIN] .NE. USE[UIW]              AIS      PROSEQ      1220     A
PROSEQ                                                                     AIS      PROSEQ      1221     A
PROSEQ   *        *MOD* ENTRY TO SAME INSTRUCTION.                         AIS      PROSEQ      1222     A
PROSEQ                                                                     AIS      PROSEQ      1223     A
PROSEQ            SA5    O.MOD                                             AIS      PROSEQ      1224     A
PROSEQ            LX4    ML.IIP-ML.MTIP                                    AIS      PROSEQ      1225     A
PROSEQ            IX7    X4+X5                                             AIS      PROSEQ      1226     A
PROSEQ            SA3    X7          MD = O.MOD + MTI[MIN] ; MDW = [MD]    AIS      PROSEQ      1227     A
PROSEQ            SB6    60                                                AIS      PROSEQ      1228     A
PROSEQ            UX1    B5,X1       RC = SC[UIW]                          AIS      PROSEQ      1229     A
PROSEQ            SB6    B6-B5                                             AIS      PROSEQ      1230     A
PROSEQ            LX3    B6          RX[Q] # RN[SHIFT(Q,-RC)]              .NO.ID.  PROSEQ      1231     A
PROSEQ            BX4    X0*X3                                             AIS      PROSEQ      1232     A
PROSEQ            BX7    X4+X6       RX[MDW] = RIM                         AIS      PROSEQ      1233     A
PROSEQ            LX7    B5                                                AIS      PROSEQ      1234     A
PROSEQ            SA7    A3                                                AIS      PROSEQ      1235     A
PROSEQ            EQ     IUM1                                              AIS      PROSEQ      1236     A
PROSEQ                                                                     AIS      PROSEQ      1237     A
PROSEQ   *        NO MATCH.  CREATE NEW *MOD* ENTRY.                       AIS      PROSEQ      1238     A
PROSEQ                                                                     AIS      PROSEQ      1239     A
PROSEQ    IUM3    SA4    A2+2        DI = R1I + 2 ;  DIW = [DI]            AIS      PROSEQ      1240     A
PROSEQ            LX4    58-D.TYP                                          AIS      PROSEQ      1241     A
PROSEQ            MI     X4,IUM4     IF TY[DIW] .GT. 2                     AIS      PROSEQ      1242     A
PROSEQ            SA4    IUMA                                              AIS      PROSEQ      1243          I
          -FCC0508
PROSEQ            IX2    X2+X4       RJ[R1W] = RJ[R1W] + 100000B ;         AIS      PROSEQ      1244          I
          -FCC0508
PROSEQ            SA3    IUMA                                                       FCC0508       10     A
PROSEQ            IX2    X2+X3       RJ[R1W] = RJ[R1W] + 100000B ;                  FCC0508       11     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ   *                          RK[R1W] = RK[R1W] + 100000B          AIS     PROSEQ      1245     A
PROSEQ                                                                   AIS     PROSEQ      1246     A
PROSEQ    IUM4     SA5    L.MOD                                          AIS     PROSEQ      1247     A
PROSEQ             BX4    X0*X2                                          AIS     PROSEQ      1248          I
          -FCC0508
PROSEQ             LX4    59-58+D.TYP-D.^DP                                      FCC0508       12     A
PROSEQ             BX7    X5          RII = L.MOD                                FCC0508       13     A
PROSEQ             PL     X4,IUM5     IF D[DIW]                                  FCC0508       14     A
PROSEQ             BX7    -X0*X2                                                 FCC0508       15     A
PROSEQ             SX7    X7+100000B  RII = RI[R1W] + 100000B                    FCC0508       16     A
PROSEQ                                                                          FCC0508       17     A
PROSEQ    IUM5     BX4    X0*X2                                                  FCC0508       18     A
PROSEQ             SB6    60                                             AIS     PROSEQ      1249     A
PROSEQ             UX1    B5,X1       RC = SC[UIW]                       .NO.ID. PROSEQ      1250     A
PROSEQ             BX7    X4+X5       RI[R1W] = L.MOD                    AIS     PROSEQ      1251          I
          -FCC0508
PROSEQ             BX7    X4+X7       RI[R1W] = RII                              FCC0508       19     A
PROSEQ             SB6    B6-B5                                          AIS     PROSEQ      1252     A
PROSEQ             LX7    B6                                             AIS     PROSEQ      1253     A
PROSEQ             BX4    X0*X7                                          AIS     PROSEQ      1254     A
PROSEQ             BX6    X4+X6       RX[R1W] = RIM                      AIS     PROSEQ      1255     A
PROSEQ             LX6    B5                                             AIS     PROSEQ      1256     A
PROSEQ             SA2    A2+B1       R2I = R1I + 1                      AIS     PROSEQ      1257     A
PROSEQ             SX4    100000B                                        AIS     PROSEQ      1258     A
PROSEQ             LX4    IH.RFP                                         AIS     PROSEQ      1259     A
PROSEQ             BX7    X2+X4                                          AIS     PROSEQ      1260     A
PROSEQ             SA6    B4                                             AIS     PROSEQ      1261     A
PROSEQ             SA7    B4+B1                                          AIS     PROSEQ      1262     A
PROSEQ             SA2    A2+B1       DI = R2I + 1                       AIS     PROSEQ      1263     A
PROSEQ             MX7    0                                              AIS     PROSEQ      1264     A
PROSEQ             BX6    X2                                             .NO.ID. PROSEQ      1265     A
PROSEQ             SA6    A7+B1                                          AIS     PROSEQ      1266     A
PROSEQ             SA7    A6+B1                                          AIS     PROSEQ      1267     A
PROSEQ             SB4    A7+B1                                          AIS     PROSEQ      1268     A
PROSEQ             SX7    X5+4                                           AIS     PROSEQ      1269     A
PROSEQ             SA7    A5          L.MOD = L.MOD + 4                  AIS     PROSEQ      1270     A
PROSEQ             SA3    L.MLT                                          AIS     PROSEQ      1271     A
PROSEQ             LX5    ML.MTIP     MTM = L.MOD - 4                    AIS     PROSEQ      1272     A
PROSEQ             MX0    1                                              AIS     PROSEQ      1273     A
PROSEQ             LX0    1+ML.DELP   DELM = 1                           AIS     PROSEQ      1274     A
PROSEQ             SA4    O.MLT                                          AIS     PROSEQ      1275     A
PROSEQ             BX5    X0+X5                                          AIS     PROSEQ      1276     A
PROSEQ             LX0    ML.NIP-ML.DELP     NIM = 1                     AIS     PROSEQ      1277     A
PROSEQ             SB5    X4                                             AIS     PROSEQ      1278     A
PROSEQ             BX6    X0+X5                                          AIS     PROSEQ      1279     A
PROSEQ             LX1    -DU.USEP                                       AIS     PROSEQ      1280     A
PROSEQ             SX5    X1                                             AIS     PROSEQ      1281     A
PROSEQ             LX5    ML.IIP      IIM = USE[UIW]                     AIS     PROSEQ      1282     A
PROSEQ             BX7    X6+X5                                          AIS     PROSEQ      1283     A
PROSEQ             SA7    X3+B5       MLT(L.MLT+1) = MLW(DELM,MTM,NIM,IIM) AIS   PROSEQ      1284     A
PROSEQ             SX6    X3+B1                                          AIS     PROSEQ      1285     A
PROSEQ             SA6    A3          L.MLT = L.MLT + 1                  AIS     PROSEQ      1286     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            EQ     IUM1                                          AIS     PROSEQ      1287     A
PROSEQ    SRI     SPACE  4,18        SRI                                       PROSEQ      1288     A
PROSEQ   **       SRI - STORE R-LIST INSTRUCTION.                      AIS     PROSEQ      1289     A
PROSEQ   *        FILE INSTRUCTIONS IN *MOD*.                          AIS     PROSEQ      1290     A
PROSEQ   *                                                             AIS     PROSEQ      1291     A
PROSEQ   *        ENTRY  X6 - R1 WORD, LESS OP-CODE.                   AIS     PROSEQ      1292     A
PROSEQ   *               X7 - R2 WORD                                  AIS     PROSEQ      1293     A
PROSEQ   *               B4 - NEXT *MOD* STORE ADDRESS.                AIS     PROSEQ      1294     A
PROSEQ   *               B5 - OP-CODE.                                 AIS     PROSEQ      1295     A
PROSEQ   *                                                             AIS     PROSEQ      1296     A
PROSEQ   *        EXIT   B4 - INCREMENTED.                             AIS     PROSEQ      1297     A
PROSEQ   *                                                             AIS     PROSEQ      1298     A
PROSEQ   *        USES   A3,A6,A7,X3                                   .NO.ID. PROSEQ      1299     A
PROSEQ                                                                 AIS     PROSEQ      1300     A
PROSEQ    SRI     ROUTINE                                                      PROSEQ      1301     A
PROSEQ            PX6    B5,X6                                         AIS     PROSEQ      1302     A
PROSEQ            SA3    =XF.RDT+B5                                    AIS     PROSEQ      1303     A
PROSEQ            SA6    B4                                            AIS     PROSEQ      1304     A
PROSEQ            SA7    B4+B1                                         AIS     PROSEQ      1305     A
PROSEQ            BX6    X3                                            AIS     PROSEQ      1306     A
PROSEQ            MX7    0                                             AIS     PROSEQ      1307     A
PROSEQ            SA6    A7+B1                                         AIS     PROSEQ      1308     A
PROSEQ            SA7    A6+B1                                         AIS     PROSEQ      1309     A
PROSEQ            SB4    A7+B1                                         AIS     PROSEQ      1310     A
PROSEQ            EQ     SRI                                           AIS     PROSEQ      1311     A
PROSEQ                                                                 AIS     PROSEQ      1312     A
PROSEQ            QUAL   *                                             AIS     PROSEQ      1313     A
PROSEQ                                                                 AIS     PROSEQ      1314     A
PROSEQ    AIS     EQU    /AIS/AIS                                      AIS     PROSEQ      1315          I
         -CCGNC
PROSEQ    AIS#    EQENT  /AIS/AIS                                              CCGNC        734     A
PROSEQ    ESR     TITLE  ESR - EXPAND SPECIAL REFERENCES                       PROSEQ      1316     A
PROSEQ    .AET    IF     DEF,WB.AETL                                           CCG5         930     A
PROSEQ                                                                         CCG5         931     A
PROSEQ   **       ESR - EXPAND SPECIAL REFERENCES.                             PROSEQ      1317     A
PROSEQ   *                                                                     PROSEQ      1318     A
PROSEQ   *        ESR EXPANDS LEVEL 2/3 AND/OR FORMAL PARAMETER MEMORY REFS    PROSEQ      1319     A
PROSEQ   *        IN A SEQUENCE.  ALL TYPE 3 LOAD, STORE AND SET INSTRUCTIONS  PROSEQ      1320     A
PROSEQ   *        WITH THE SPECIAL PROPERTIES DETERMINED BY *KEY* (X5) ARE     PROSEQ      1321     A
PROSEQ   *        REPLACED WITH INDIRECT OR EXPANDED REFS.                     PROSEQ      1322     A
PROSEQ   *                                                                     PROSEQ      1323     A
PROSEQ   *        ENTRY  X5 - KEY, EXPANSION FLAG AS FOLLOWS--                 PROSEQ      1324     A
PROSEQ   *                   1 = EXPAND F.P. REFERENCES ONLY                   PROSEQ      1325     A
PROSEQ   *                   2 = EXPAND LCM REFERENCES ONLY                    PROSEQ      1326     A
PROSEQ   *                   3 = EXPAND F.P. OR LCM REFERENCES                 PROSEQ      1327     A
PROSEQ   *                   4 = EXPAND LCM REFERENCES OR F.P. STT/S .         PROSEQ      1328     A
PROSEQ   *               X6 - SQUEEZE FLAG FOR MPB .                           PROSEQ      1329     A
PROSEQ                                                                         PROSEQ      1330          I
         -CCGNC
PROSEQ   *ESRA    VFD    60/=XN.FP                                             PROSEQ      1331          I
         -FCC0643 -CCGNC
PROSEQ            ENTRY  ESRA                                                  FCC0643       46          I

         0        1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
                -CCGNC
PROSEQ           ESRA     VFD     60/-1        MAKE FP-ONLY CALL ILLEGAL                PROSEQ      1332          I
                -CCGNC
PROSEQ                    VFD     60/=XLEVEL                                            PROSEQ      1333          I
                -FCC0504  -CCGNC
PROSEQ                    VFD     60/=XLEVEL2                                           FCC0504       19          I
                -CCGNC
PROSEQ                    VFD     60/=XFPL2                                             PROSEQ      1334          I
                -CCG1     -CCGNC
PROSEQ                    VFD     60/=XFPL2                                             PROSEQ      1335          I
                -CCG1     -CCGNC
PROSEQ                    VFD     60/=XCG.SRF                                           CCG1         193          I
                -CCGNC
PROSEQ                    VFD     60/=XCG.SRF                                           CCG1         194          I
                -CCGNC
PROSEQ                                                                                  PROSEQ      1336          I
                -CCGNC
PROSEQ                                                                                  CCGNC        735     A
PROSEQ           ESRA     ENTRY.  -1,#         MAKE FP ONLY CALL ILLEGAL                CCGNC        736     A
PROSEQ                    VFD     42/,18/=XHO$LVL2                                      CCGNC        737     A
PROSEQ                    VFD     42/,18/CC$SRF                                         CCGNC        738     A
PROSEQ                    VFD     42/,18/CC$SRF                                         CCGNC        739     A
PROSEQ                                                                                  CCGNC        740     A
PROSEQ           FSR      BSS     1            FIRST SPECIAL REF. (CHAIN HEAD)          PROSEQ      1337     A
PROSEQ           LSF      BSS     1            LOAD/STORE/SET FLAG                      PROSEQ      1338     A
PROSEQ           SQF      BSS     1            SQUEEZE FLAG                             PROSEQ      1339     A
PROSEQ           R2W      BSS     1            R2 WORD OF CURRENT INSTRUCTION           FCC0643       47     A
PROSEQ                                                                                  PROSEQ      1340     A
PROSEQ           ESR0     SA3     O.MOD                                                 PROSEQ      1341     A
PROSEQ                    SB7     X3-1                                                  PROSEQ      1342     A
PROSEQ                    SX6     A7-B7                                                 PROSEQ      1343     A
PROSEQ                    SA6     L.MOD                                                 PROSEQ      1344     A
PROSEQ                                                                                  PROSEQ      1345     A
PROSEQ           ESR1     SA1     O.SEQ                                                 PROSEQ      1346     A
PROSEQ                    SA2     X1                                                    PROSEQ      1347     A
PROSEQ                    UX7     B2,X2                                                 PROSEQ      1348     A
PROSEQ                    SA3     F.RDT+B2                                              PROSEQ      1349     A
PROSEQ                    SA4     SQF                                                   PROSEQ      1350     A
PROSEQ                    BX7     X3                                                    PROSEQ      1351     A
PROSEQ                    LX6     X4           (X6) = SQF FOR *MPB*                     PROSEQ      1352     A
PROSEQ                    SA7     X1+2         [SEQ+2] = RDT(OC[SEQ])                   PROSEQ      1353     A
PROSEQ                    MI      X6,ESR       IF SQF < 0                               PROSEQ      1354     A
PROSEQ                    CALL    MPB          MERGE MODS WITH SEQUENCE                 PROSEQ      1355          I
                -CCGNC
PROSEQ                    CALL    MPB#         MERGE MODS WITH SEQUENCE                 CCGNC        741     A
PROSEQ                                                                                  PROSEQ      1356     A
PROSEQ           ESR      ENTRY.  **                                                    PROSEQ      1357          I
                -CCGNC
PROSEQ           ESR      ENTRY.  **,#                                                  CCGNC        742     A
PROSEQ                    SA4     ESRA-1+X5                                             PROSEQ      1358     A
PROSEQ                    SA6     SQF                                                   PROSEQ      1359     A
PROSEQ                    SA3     X4                                                    PROSEQ      1360     A

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            ZR     X3,ESR1     IF FLAG(MODE) = 0  */ NO SPECIAL REFS      PROSEQ    1361    A
PROSEQ            SA1    TSBA+X5     MASK FOR SR BITS                           PROSEQ    1362    A
PROSEQ            SA4    O.SEQ                                                  PROSEQ    1363    A
PROSEQ            MX7    0                                                      PROSEQ    1364    A
PROSEQ            BX0    X1                                                     PROSEQ    1365    A
PROSEQ            SA7    X4+2        LOOP TERMINATOR AT D(BOS)                  PROSEQ    1366    A
PROSEQ            RJ     MSR         MARK SPECIAL REFS                          PROSEQ    1367    A
PROSEQ            ZR     B2,ESR1     IF N.SR = 0                                PROSEQ    1368    A
PROSEQ                                                                          PROSEQ    1369    A
PROSEQ            ALLOC  MLT,B2      ALLOC( MLT , N.SR )                        PROSEQ    1370    A
PROSEQ            BX2    X1                                                     PROSEQ    1371    A
PROSEQ            LX1    4                                                      PROSEQ    1372    A
PROSEQ            LX2    2                                                      PROSEQ    1373    A
PROSEQ            IX1    X1+X2                                                  PROSEQ    1374    A
PROSEQ            ALLOC  MOD,X1      ALLOC (MOD, 5*4*N.SR )                     PROSEQ    1375    A
PROSEQ                                                                          PROSEQ    1376    A
PROSEQ            RJ     GSR         GENERATE SPECIAL REFS                      PROSEQ    1377    A
PROSEQ            EQ     ESR0                                                   PROSEQ    1378    A
PROSEQ    MSR     SPACE  4,8         MSR                                        PROSEQ    1379          I
          -FCC0643
PROSEQ    MSR     SPACE  3,14                                                   FCC0643     48    A
PROSEQ   **       MSR - MARK SPECIAL REFERENCES.                                PROSEQ    1380    A
PROSEQ   *        ENTRY  A4,X4 - O.SEQ                                          PROSEQ    1381    A
PROSEQ   *               X0 - MASK OF PERTINENT SPECIAL REF BITS                PROSEQ    1382    A
PROSEQ   *                                                                      PROSEQ    1383    A
PROSEQ   *        EXIT   B2 = N.SR                                              PROSEQ    1384    A
PROSEQ   *               B3 = (FSR) = POINTS TO S.R. CHAIN IN LINK WORDS        PROSEQ    1385    A
PROSEQ                                                                          PROSEQ    1386    A
PROSEQ    MSRA    BSS    0                                                      PROSEQ    1387    A
PROSEQ            LOC    0                                                      PROSEQ    1388    A
PROSEQ            CON    0                                                      PROSEQ    1389    A
PROSEQ            CON    0                                                      PROSEQ    1390    A
PROSEQ            CON    0                                                      PROSEQ    1391    A
PROSEQ            BFMW   D,(LD,ST)                                              PROSEQ    1392    A
PROSEQ            LOC    *O                                                     PROSEQ    1393    A
PROSEQ                                                                          PROSEQ    1394    A
PROSEQ    MSR0    SX6    B3                                                     PROSEQ    1395    A
PROSEQ            SA6    FSR         FIRST SPECIAL REF LINK ADDRESS             PROSEQ    1396    A
PROSEQ                                                                          PROSEQ    1397    A
PROSEQ    MSR     ROUTINE            **ENTRY/EXIT**                             PROSEQ    1398    A
PROSEQ            SA5    MSRA-1+X5                                              PROSEQ    1399    A
PROSEQ            SA2    X4+B1                                                  PROSEQ    1400    A
PROSEQ            SX1    X2          LEN = TXTL[SEQ+1]                          PROSEQ    1401    A
PROSEQ            SB2    B0          N.SR = 0                                   PROSEQ    1402    A
PROSEQ            IX6    X4+X1                                                  PROSEQ    1403    A
PROSEQ            SB7    X6-2        D(EOQ), I=N.INS                            PROSEQ    1404    A
PROSEQ            SA3    B7          DI=D(I)                                    PROSEQ    1405    A
PROSEQ            SB3    B0          CP = 0                                     PROSEQ    1406    A
PROSEQ            SB6    X4                                                     PROSEQ    1407    A
PROSEQ            SX7    B1                                                     PROSEQ    1408    A
PROSEQ            LX7    D.FPP                                                  PROSEQ    1409    A
PROSEQ                                                                          PROSEQ    1410    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ    MSR1     ZR     X3,MSR0     IF DI = 0     */ I = 0                    PROSEQ    1411    A
PROSEQ             SB7    B7-4        I = I - 1                                 PROSEQ    1412    A
PROSEQ             BX4    X0*X3       SRBITS(I+1)                               PROSEQ    1413    A
PROSEQ             BX2    X5*X3                                                 PROSEQ    1414    A
PROSEQ             BX1    X2                                                    PROSEQ    1415    A
PROSEQ             LX2    D.FPP-D.LDP                                           PROSEQ    1416    A
PROSEQ             LX1    D.FPP-D.STP                                           PROSEQ    1417    A
PROSEQ             BX2    X2+X1       LDSTI = MSRA(KEY).AND. (LD[DI] .OR. ST[DI])  PROSEQ  1418    A
PROSEQ             BX1    X7*X2                                                 PROSEQ    1419    A
PROSEQ             BX4    -X1*X4      FP[SR[DI]] = FP[SR[DI]] .AND. ^LDSTI      PROSEQ    1420    A
PROSEQ             SA3    B7          DI=D(I)                                   PROSEQ    1421    A
PROSEQ             ZR     X4,MSR1     IF NOT SPECIAL REF(MODE)                  PROSEQ    1422    A
PROSEQ             SB2    B2+B1       N.SR = N.SR + 1                           PROSEQ    1423    A
PROSEQ             SX6    B3                                                    PROSEQ    1424    A
PROSEQ             SA6    B7+5        (LINK(I-1)) = CP                          PROSEQ    1425    A
PROSEQ             SB3    A6-B6       CP = LINK(I-1) - O.SEQ                    PROSEQ    1426    A
PROSEQ             EQ     MSR1                                                  PROSEQ    1427    A
PROSEQ    GSR      SPACE  4,8         GSR                                       PROSEQ    1428    A
PROSEQ   **        GSR - GENERATE SPECIAL REFERENCES.                           PROSEQ    1429    A
PROSEQ   *                                                                      PROSEQ    1430    A
PROSEQ   *         ENTRY  B2 = CHAIN HEAD FOR SPECIAL REFS                      PROSEQ    1431    A
PROSEQ   *                                                                      PROSEQ    1432    A
PROSEQ    GSR      ROUTINE            **ENTRY/EXIT**                            PROSEQ    1433    A
PROSEQ             SA2    FSR                                                   PROSEQ    1434    A
PROSEQ             SA1    O.SEQ                                                 PROSEQ    1435    A
PROSEQ             MX7    0                                                     PROSEQ    1436    A
PROSEQ             SB7    X1                                                    PROSEQ    1437    A
PROSEQ             SB2    X2+B7                                                 PROSEQ    1438    A
PROSEQ             SB3    100000B     OFFSET FOR MPB                            PROSEQ    1439    A
PROSEQ             SA1    O.MOD                                                 PROSEQ    1440    A
PROSEQ             SA7    L.MLT       L.MLT = 0                                 PROSEQ    1441    A
PROSEQ             SA7    X1+3        A7= FWA - 1 EXPANSION IN MOD              PROSEQ    1442    A
PROSEQ             SB4    4           RI = 4                                    PROSEQ    1443    A
PROSEQ             EQ     GSR3                                                  PROSEQ    1444    A
PROSEQ                                                                          PROSEQ    1445    A
PROSEQ    GSR1     RJ     ISC         ISSUE SET CODE                            PROSEQ    1446    A
PROSEQ                                                                          PROSEQ    1447    A
PROSEQ    GSR2     RJ     TRE         TERMINATE REF EXPANSION                   PROSEQ    1448    A
PROSEQ             SA1    O.SEQ                                                 PROSEQ    1449    A
PROSEQ             SB7    X1+                                                   PROSEQ    1450    A
PROSEQ                                                                          PROSEQ    1451    A
PROSEQ    GSR3     EQ     B2,B7,GSR   IF END OF CHAIN                           PROSEQ    1452    A
PROSEQ             SA0    A7          IMA = INITIAL MOD ADDRESS - 1             PROSEQ    1453    A
PROSEQ             SA1    B2-B1       DI = D(I)                                 PROSEQ    1454    A
PROSEQ             SA3    B2                                                    PROSEQ    1455    A
PROSEQ             SA2    A1-B1       R2I = DI - 1                              FCC0643     49    A
PROSEQ             BX6    X2          R2W = [R2I]                               FCC0643     50    A
PROSEQ             SA6    R2W                                                   FCC0643     51    A
PROSEQ             MX2    -D.LDSTL                                              PROSEQ    1456          I
          -CCG35
PROSEQ             BX4    X1                                        001550      CCG35       23          I
          -CCGA040

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            LX4    -D.XMTP                                          001560   CCG35        24          I
          -CCGA040
PROSEQ            SX2    B1                                               001570   CCG35        25          I
          -CCGA040
PROSEQ            BX6    X2*X4                                            001580   CCG35        26          I
          -CCGA040
PROSEQ            LX6    D.L2P                                            001590   CCG35        27          I
          -CCGA040
PROSEQ            SA6    =SLEV0      LEV0 = XMT[DI]  */  =1 IF LEVEL = 0 001600   CCG35        28          I
          -CCGA040
PROSEQ   *                                                                        CCGA040       5     A
PROSEQ   *        EXAMINE THE SYMBOL TABLE AND DETERMINE IF REFERENCED            CCGA040       6     A
PROSEQ   *        SYMBOL IS LEVEL 0                                               CCGA040       7     A
PROSEQ   *                                                                        CCGA040       8     A
PROSEQ            MX6    0                                                       CCGA040       9     A
PROSEQ            SB6    I.GL                                                    CCGA040      10     A
PROSEQ            SB5    X2                                                      CCGA040      11     A
PROSEQ            ZR     B5,GSR3A    IF IH[R2] = 0                               CCGA040      12     A
PROSEQ            GE     B5,B6,GSR3A       IF I[R2] " 0                          CCGA040      13     A
PROSEQ            BX5    X1                                                      CCGA040      14     A
PROSEQ            LX5    59-D.TYP                                                CCGA040      15     A
PROSEQ            MI     X5,GSR3A    IF TYPE[DI] = II & IV                       CCGA040      16     A
PROSEQ            SA5    =X0.SYM                                                 CCGA040      17     A
PROSEQ            SX4    B5+B5                                                   CCGA040      18     A
PROSEQ            SB6    X5+B5                                                   CCGA040      19     A
PROSEQ            SX5    X4+B6                                                   CCGA040      20     A
PROSEQ            SA4    X5+B1       WORDB = SYM(3*IH+1)                         CCGA040      21     A
PROSEQ            LX4    59-WB.LABP                                              CCGA040      22     A
PROSEQ            MI     X4,GSR3A    IF LAB[WORDB]                               CCGA040      23     A
PROSEQ            LX4    WB.LABP-WB.LEVNP+1                                      CCGA040      24     A
PROSEQ            MX5    -WB.LEVNL                                               CCGA040      25     A
PROSEQ            BX5    -X5*X4                                                  CCGA040      26     A
PROSEQ            NZ     X5,GSR3A    IF LEVN[WB] NE 0  */LEVEL 0                 CCGA040      27     A
PROSEQ            LX4    WB.LEVNP-WB.LCMP                                        CCGA040      28     A
PROSEQ            SX6    B1                                                      CCGA040      29     A
PROSEQ            BX6    X4*X6       IFF LCM[WORDB]                              CCGA040      30     A
PROSEQ            LX6    D.L2P                                                   CCGA040      31     A
PROSEQ   *                                                                        CCGA040      32     A
PROSEQ    GSR3A   SA6    =SLEV0                                                  CCGA040      33     A
PROSEQ            MX2    -D.LDSTL                                         001610 CCG35        29     A
PROSEQ            SB2    X3+B7       I = I + 1                                   PROSEQ     1457     A
PROSEQ            BX3    X1                                                      PROSEQ     1458     A
PROSEQ            LX3    -D.LDSTP                                                PROSEQ     1459     A
PROSEQ            BX2    -X2*X3                                                  PROSEQ     1460     A
PROSEQ            BX6    -X2                                                     PROSEQ     1461     A
PROSEQ            SA6    LSF         LOAD/STORE/SET FLAG                         PROSEQ     1462     A
PROSEQ            ZR     X6,GSR1     IF NOT LOAD OR STORE                        PROSEQ     1463     A
PROSEQ                                                                           PROSEQ     1464     A
PROSEQ    #DAL    IFNE   .DAL,0      IF LEVEL 2 ALLOWED                          PROSEQ     1465     A
PROSEQ            BX2    X0*X1                                                   PROSEQ     1466     A
PROSEQ            LX2    59-D.L2P                                                PROSEQ     1467     A
PROSEQ            PL     X2,GSR4     IF NOT LCM REF                              PROSEQ     1468     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            RJ     ISC         ISSUE SET                                  PROSEQ      1469     A
PROSEQ            RJ     IDR         ISSUE DIRECT LCM REF                       PROSEQ      1470     A
PROSEQ            EQ     GSR2                                                   PROSEQ      1471     A
PROSEQ                                                                          PROSEQ      1472     A
PROSEQ    GSR4    BSS    0                                                      PROSEQ      1473     A
PROSEQ    #DAL    ENDIF                                                         PROSEQ      1474     A
PROSEQ                                                                          PROSEQ      1475     A
PROSEQ            RJ     IAL         ISSUE ADDRESS LOAD                         PROSEQ      1476     A
PROSEQ            RJ     IRA         ISSUE RA FOR RF                            PROSEQ      1477     A
PROSEQ            RJ     IPR         ISSUE PLD/PST                              PROSEQ      1478     A
PROSEQ            EQ     GSR2                                                   PROSEQ      1479     A
PROSEQ    ISC     EJECT              ISC                                        PROSEQ      1480     A
PROSEQ   **       ISC - ISSUE SET CODE.                                         PROSEQ      1481     A
PROSEQ   *                                                                      PROSEQ      1482     A
PROSEQ   *        ISC CONTROLS EXPANSION OF SPECIAL REFERENCES FOR STT INSTRUC- PROSEQ      1483     A
PROSEQ   *        TIONS.  ALL STT TO FP OR LCM SYMBOLS ARE EXPANDED AS DESIRED. PROSEQ      1484     A
PROSEQ   *        IN ADDITION, ISC GENERATES ALL CODE PRECEDING THE DRL/DWL     PROSEQ      1485     A
PROSEQ   *        FOR THE EXPANSION OF A LEVEL 2 LOAD OR STORE.  ISC INSURES    PROSEQ      1486     A
PROSEQ   *        THAT AN XMT IS GENERATED FOR ANY STT (FP OR LCM=I MODE LCM)   PROSEQ      1487     A
PROSEQ   *        IN WHICH THE LAST OPERATION IS A LOAD.  THIS GUARANTEES THAT  PROSEQ      1488     A
PROSEQ   *        A STORE-TO-APLIST MAY FOLLOW.                                 PROSEQ      1489     A
PROSEQ   *                                                                      PROSEQ      1490     A
PROSEQ   *        ENTRY  A1 = ADDRESS OF DESCRIPTOR OF CURRENT SPECIAL REF.     PROSEQ      1491     A
PROSEQ   *               A7 = FWA-1 OF MOD BUFFER TO BE USED FOR EXPANSION.     PROSEQ      1492     A
PROSEQ   *               X1 = DESCRIPTOR                                        PROSEQ      1493     A
PROSEQ   *               X0 = MASK OF SPECIAL (L2 AND/OR FP) BITS DESCRIBING    PROSEQ      1494     A
PROSEQ   *                    THE TYPES OF REFS TO BE EXPANDED.                 PROSEQ      1495     A
PROSEQ   *               B1 = 1                                                 PROSEQ      1496     A
PROSEQ   *               B3 = 100000B                                           PROSEQ      1497     A
PROSEQ   *               B4 = NEXT RI NUMBER (UPDATED BY SRI)                   PROSEQ      1498     A
PROSEQ   *                                                                      PROSEQ      1499     A
PROSEQ   *        EXIT   A0,A1,B1,B2,B3,X0 PRESERVED.                           PROSEQ      1500     A
PROSEQ   *               B4 AS LAST UPDATED BY SRI.                             PROSEQ      1501     A
PROSEQ   *               EXPANSION FILED IN MOD, WITH A7 UPDATED.               PROSEQ      1502     A
PROSEQ   *                                                                      PROSEQ      1503     A
PROSEQ   *        CALLS  SRI, IAL, IRA, ISX.                                    PROSEQ      1504     A
PROSEQ                                                                          CCGA044        9          I
          -CCGA052
PROSEQ    CHK18   CON    400000B     CHECKS SIZE OF CA                          CCGA044       10          I
          -CCGA052
PROSEQ                                                                          PROSEQ      1505     A
PROSEQ    ISC0    MX2    0                                                      PROSEQ      1506     A
PROSEQ            RJ     SRI         STORE INSTRUCTION                          PROSEQ      1507     A
PROSEQ                                                                          PROSEQ      1508     A
PROSEQ    ISC     ROUTINE            **ENTRY/EXIT**                             PROSEQ      1509     A
PROSEQ            BX3    X0*X1                                                  PROSEQ      1510     A
PROSEQ            LX3    59-D.L2P                                               PROSEQ      1511     A
PROSEQ            MI     X3,ISC3     IF LCM REF                                 PROSEQ      1512     A
PROSEQ            LX3    D.L2P-D.FPP                                            PROSEQ      1513     A
PROSEQ            PL     X3,ISC      IF NOT FP                                  PROSEQ      1514     A
PROSEQ                                                                          PROSEQ      1515     A
PROSEQ    ISC2    RJ     IAL         ISSUE ADDRESS LOAD                         PROSEQ      1516     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            RJ    IRA         ISSUE IA FOR ANY RF                          PROSEQ     1517    A
PROSEQ            RJ    ISX         ISSUE STT/XMT                                PROSEQ     1518    A
PROSEQ            EQ    ISC                                                      PROSEQ     1519    A
PROSEQ                                                                           PROSEQ     1520    A
PROSEQ   ISC3     SA4   =XDIRECT                                                 PROSEQ     1521        I
        -CCG1
PROSEQ   ISC3     SA4   =XCO.LCM                                                 CCG1        195        I
        -CCGNC
PROSEQ   ISC3     SA4   =XHO$LCM                                                 CCGNC       743    A
PROSEQ            NZ    X4,ISC5     IF LCM=I MODE                                PROSEQ     1522    A
PROSEQ            SA4   LEV0                                          001630     CCG35        30    A
PROSEQ            NZ    X4,ISC5 IF LEVEL = 0                          001640     CCG35        31    A
PROSEQ                                                                           PROSEQ     1523    A
PROSEQ   #DAL     IFNE  .DAL,0      IF LEVEL 2 ALLOWED                           PROSEQ     1524    A
PROSEQ            SA4   LSF                                                      PROSEQ     1525    A
PROSEQ            ZR    X4,ISC1     IF  STT LCM                                  PROSEQ     1526    A
PROSEQ            LX3   D.L2P-D.FPP                                              PROSEQ     1527    A
PROSEQ            MI    X3,ISC2     IF FP                                        PROSEQ     1528    A
PROSEQ            RJ    ISI         ISSUE STT INSTRUCTIIN                        PROSEQ     1529    A
PROSEQ            EQ    ISC                                                      PROSEQ     1530    A
PROSEQ                                                                           PROSEQ     1531    A
PROSEQ   #DAL     ENDIF                                                          PROSEQ     1532    A
PROSEQ                                                                           PROSEQ     1533    A
PROSEQ   ISC1     LX3   D.L2P-D.FPP                                              PROSEQ     1534    A
PROSEQ            MI    X3,ISC2     IF FP                                        PROSEQ     1535    A
PROSEQ            RJ    ISI         ISSUE STT                                    PROSEQ     1536    A
PROSEQ            RJ    IMO         ISSUE MASK-OR BIT 59                         PROSEQ     1537    A
PROSEQ            EQ    ISC                                                      PROSEQ     1538    A
PROSEQ                                                                           PROSEQ     1539    A
PROSEQ   *        IN LCM=I MODE, ALL LCM REFS ARE EXPANDED IRREGARDLESS OF       PROSEQ     1540        I
        -CCGA044
PROSEQ   *        IN LCM=I MODE, ALL LCM REFS ARE EXPANDED REGARDLESS OF         CCGA044      11    A
PROSEQ   *        EXPANSION MODE BECAUSE OF THE ODD MEANING OF THE IH SYMBOL     PROSEQ     1541    A
PROSEQ   *        (SCM POINTER CELL ADDRESS).                                    PROSEQ     1542    A
PROSEQ                                                                           PROSEQ     1543    A
PROSEQ   ISC5     LX1   59-D.FPP    FP BIT IRREGARDLESS OF MODE                  PROSEQ     1544    A
PROSEQ            PL    X1,ISC6     IF NOT FP                                    PROSEQ     1545    A
PROSEQ            RJ    IAL         ISSUE ADDRESS LOAD                           PROSEQ     1546    A
PROSEQ            MX5   0           RA = 0                                       CCG18       113    A
PROSEQ            EQ    ISC7                                                     PROSEQ     1547    A
PROSEQ                                                                           PROSEQ     1548    A
PROSEQ   *        GENERATE CODE TO LOAD SCM POINTER CELL THAT CONTAINS 21-BIT    PROSEQ     1549    A
PROSEQ   *        RELOCATED LCM ADDRESS AT OBJECT TIME.                          PROSEQ     1550    A
PROSEQ                                                                           PROSEQ     1551    A
PROSEQ   ISC6     SA2   A1-B1       IHINFO(REF)                                  PROSEQ     1552    A
PROSEQ            LX2   -IH.IHP                                                  PROSEQ     1553    A
PROSEQ            SX7   X2                                                       PROSEQ     1554        I
        -FCC0882
PROSEQ            SB5   OC.LD                                                    PROSEQ     1555        I
        -FCC0882
PROSEQ            LX7   IH.IHP                                                   PROSEQ     1556        I
        -FCC0882

        0         1         2         3         4         5         6         7         8
        12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            SX6    B4          RI                                          PROSEQ      1557          I
          -FCC0645  -FCC0882
PROSEQ            LX6    R1.RIP                                                  PROSEQ      1558          I
          -FCC0645  -FCC0882
PROSEQ            SX2    B1                                                      FCC0645       33          I
          -FCC0882
PROSEQ            SX7    X2+IH.LCMA  LCMA[IH] = 1  */ INDICATE ADDRESS LOAD      FCC0882       11          I
          -CCG18
PROSEQ            SA3    =XO$SYM                                                 CCG18        114     A
PROSEQ            SX7    X2                                                      CCG18        115     A
PROSEQ            LX2    B1,X7                                                   CCG18        116     A
PROSEQ            IX2    X7+X2                                                   CCG18        117     A
PROSEQ            IX7    X3+X2                                                   CCG18        118     A
PROSEQ            SA3    X7+2                                                    CCG18        119     A
PROSEQ            MX2    -WC.RBL                                                 CCG18        120     A
PROSEQ            LX3    -WC.RBP                                                 CCG18        121     A
PROSEQ            BX7    -X2*X3                                                  CCG18        122     A
PROSEQ            MX2    -WC.RAL                                                 CCG18        123     A
PROSEQ            LX3    WC.RBP-WC.RAP                                           CCG18        124     A
PROSEQ            BX5    -X2*X3                                                  CCG18        125     A
PROSEQ            SA2    =XO$CBT                                                 CCG18        126     A
PROSEQ            IX3    X2+X7                                                   CCG18        127     A
PROSEQ            SA2    X3+CB.W                                                 CCG18        128     A
PROSEQ            MX3    -CB.TAGL                                                CCG18        129     A
PROSEQ            LX2    -CB.TAGP                                                CCG18        130     A
PROSEQ            BX7    -X3*X2      IH = TAG[CBT(RB[WC(REF)])]                  CCG18        131     A
PROSEQ            SX2    B1                                                      FCC0882       12     A
PROSEQ            SB5    OC.LD                                                   FCC0882       13     A
PROSEQ            SX6    B4          RI                                          FCC0645       34     A
PROSEQ            LX2    R1.INP+1    IN[R1W] = 2   */ UDI = 1 ( LOAD ONLY )      FCC0645       35     A
PROSEQ            BX6    X2+X6                                                   FCC0645       36     A
PROSEQ            MX2    0                                                       PROSEQ      1559     A
PROSEQ            RJ     SRI         ISSUE LD LCM POINTER CELL                   PROSEQ      1560     A
PROSEQ                                                                           PROSEQ      1561     A
PROSEQ    ISC7    SA2    A1-B1       IHINFO(REF)                                 PROSEQ      1562     A
PROSEQ            LX2    -IH.RFP                                                 PROSEQ      1563     A
PROSEQ            MX3    -IH.RFL                                                 PROSEQ      1564     A
PROSEQ            BX4    -X3*X2                                                  PROSEQ      1565     A
PROSEQ            MX1    -IH.CAL                                                 PROSEQ      1566          I
          -CCGA044
PROSEQ            MX1    -WC.RAL                                                 CCGA044       12          I
          -CCGA052
PROSEQ            LX2    IH.RFP-IH.CAP                                           PROSEQ      1567     A
PROSEQ            BX3    -X1*X2                                                  PROSEQ      1568          I
          -CCG18
PROSEQ            SX3    X2                                                      CCG18        132     A
PROSEQ   ********TEMP. BEWARE OF 24 BIT CA.                                      CCG18        133          I
          -CCG35
PROSEQ            IX3    X5+X3       CA = CA + RA                                CCG18        134     A
PROSEQ            BX3    -X1*X3                                                  CCG18        135          I
          -CCGA052
PROSEQ            BX2    X3+X4                                                   PROSEQ      1569     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            MI     X2,ISC8     IF RF(REF) .OR. CA(REF) .NE. 0                    CCG0085       5     A
PROSEQ            NZ     X2,ISC8     IF RF(REF) .OR. CA(REF) .NE. 0                    PROSEQ      1570     A
PROSEQ            SA4    LSF                                                           PROSEQ      1571     A
PROSEQ            NZ     X4,ISC      IF LD/ST                                          PROSEQ      1572     A
PROSEQ            MX1    -R1.RIL                                                       PROSEQ      1573     A
PROSEQ            SA2    A2-B1       R1(REF)                                           PROSEQ      1574     A
PROSEQ            LX2    -R1.RIP                                                       PROSEQ      1575     A
PROSEQ            BX3    -X1*X2                                                        PROSEQ      1576     A
PROSEQ            SX6    X3+B3       RI MOD TO SEQ                                     PROSEQ      1577     A
PROSEQ            SX2    B4-4        RJ = RI(PRED)                                     PROSEQ      1578     A
PROSEQ            RJ     IXI         ISSUE AN *XMT/SA*                                 PROSEQ      1579     A
PROSEQ            EQ     ISC                                                           PROSEQ      1580     A
PROSEQ                                                                                PROSEQ      1581     A
PROSEQ    ISC8    SX6    B4          RI                                                PROSEQ      1582     A
PROSEQ            SA2    =XHO$LCM                                             001670   CCG35         32           I
          -CCGA044
PROSEQ            LX6    R1.RIP                                                        PROSEQ      1583           I
          -FCC1353
PROSEQ            ZR     X4,ISC9     IF RF .EQ. 0                                      PROSEQ      1584           I
          -FCC1353
PROSEQ            SX4    X4+B3       RF(REF) + 100000B                                 PROSEQ      1585           I
          -FCC1353
PROSEQ            ZR     X3,ISC11    IF CA .EQ. 0  RK = RF                             PROSEQ      1586           I
          -FCC1353
PROSEQ            SB5    OC.STT                                                        PROSEQ      1587           I
          -FCC1353
PROSEQ            ZR     X4,ISC9     IF RF = 0                                         FCC1353        5     A
PROSEQ            SA5    B7+X4       R1P = SEQ(RF)                                     FCC1353        6     A
PROSEQ            SX4    X4+B3       RF = RF + 100000B                                 FCC1353        7     A
PROSEQ            NZ     X3,ISC8A    IF CA " 0                                         FCC1353        8     A
PROSEQ                                                                                FCC1353        9     A
PROSEQ            UX5    B5,X5                                                         FCC1353       10     A
PROSEQ            SX7    B5-OC.DEF                                                     FCC1353       11     A
PROSEQ            NZ     X7,ISC11    IF OC[R1P] " OC.DEF                               FCC1353       12     A
PROSEQ            LX5    58-R1.SOP-SO.RTP                                              FCC1353       13     A
PROSEQ            MI     X5,ISC11    IF SORT[R1P] = 2  */ OPERAND IN A X-REGISTER      FCC1353       14     A
PROSEQ                                                                                FCC1353       15     A
PROSEQ            LX4    R1.RJP                                                        FCC1353       16     A
PROSEQ            BX6    X6+X4                                                         FCC1353       17     A
PROSEQ            MX7    0                                                             FCC1353       18     A
PROSEQ            SB5    OC.SA                                                         FCC1353       19     A
PROSEQ            EQ     ISC10                                                         FCC1353       20     A
PROSEQ                                                                                FCC1353       21     A
PROSEQ    ISC8A   SB5    OC.STT                                                        FCC1353       22           I
          -CCG35
PROSEQ    ISC8A   MI     X2,ISC8B    IF LCM = G                               001690   CCG35         33           I
          -CCGA044
PROSEQ    *       CHECK IF CA CAN BE CONTAINED IN AN 18 BIT FIELD.                     CCGA044       13     A
PROSEQ    *       IF NOT THEN WE MUST MATERIALIZE THE CONSTANT IN THE                  CCGA044       14     A
PROSEQ    *       CONSTANT TABLE                                                       CCGA044       15     A
PROSEQ                                                                                CCGA044       16     A
PROSEQ    ISC8A   SA2    CHK18                                                         CCGA044       17           I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

          -CCGA052
PROSEQ              IX2    X2-X3                                          CCGA044       18          I
          -CCGA052
PROSEQ              PL     X2,ISC8B    IF CA < CHK18                      CCGA044       19          I
          -CCGA052
PROSEQ    ISC8A     BX2    X3                                             CCGA052        9     A
PROSEQ              AX2    17                                             CCGA052       10     A
PROSEQ              ZR     X2,ISC8B    IF CA FITS IN 18 BITS              CCGA052       11     A
PROSEQ              RJ     ILC                                            CCGA044       20     A
PROSEQ              EQ     ISC8D                                          CCGA044       21     A
PROSEQ                                                                    CCGA044       22     A
PROSEQ    ISC8B     SA2    =XHO$LCM                                       CCGA044       23          I
          -CCGA052
PROSEQ    ISC8B     MX7    -IH.CAL                                        CCGA052       12     A
PROSEQ              BX3    -X7*X3      TRUNCATE NEGAITVE CA S TO 18 BITS  CCGA052       13     A
PROSEQ              SA2    =XHO$LCM                                       CCGA052       14     A
PROSEQ              MI     X2,ISC8C    IF LCM = G                         CCGA044       24          I
          -CCGA099
PROSEQ              NZ     X2,ISC8C    IF LCM = I OR LCM = G              CCGA099       29     A
PROSEQ              SB5    OC.STT                                 001700  CCG35         34     A
PROSEQ              LX4    IH.RFP                                         PROSEQ      1588     A
PROSEQ              LX3    IH.CAP                                         PROSEQ      1589     A
PROSEQ              BX7    X4+X3                                          PROSEQ      1590     A
PROSEQ              EQ     ISC10                                          PROSEQ      1591     A
PROSEQ                                                         001720     CCG35         35     A
PROSEQ    ISC8B     SB5    OC.S                                   001730  CCG35         36          I
          -CCGA044
PROSEQ    ISC8C     SB5    OC.S                                           CCGA044       25     A
PROSEQ              MX7    0                                      001740  CCG35         37     A
PROSEQ              LX3    R1.INP                                 001750  CCG35         38     A
PROSEQ              BX6    X6+X3                                  001760  CCG35         39     A
PROSEQ              MX2    0                                      001770  CCG35         40     A
PROSEQ              RJ     SRI                                    001780  CCG35         41     A
PROSEQ              SX6    B4                                     001790  CCG35         42          I
          -CCGA044
PROSEQ                                                                    CCGA044       26     A
PROSEQ    ISC8D     SX6    B4                                             CCGA044       27     A
PROSEQ              SB5    OC.IA                                  001800  CCG35         43     A
PROSEQ              SX7    B4-4                                   001810  CCG35         44     A
PROSEQ              LX7    R1.RJP                                 001820  CCG35         45     A
PROSEQ              LX4    R1.RKP                                 001830  CCG35         46     A
PROSEQ              BX7    X7+X4                                  001840  CCG35         47     A
PROSEQ              BX6    X6+X7                                  001850  CCG35         48     A
PROSEQ              MX7    0                                      001860  CCG35         49     A
PROSEQ              MX2    0                                      001870  CCG35         50     A
PROSEQ              RJ     SRI                                    001880  CCG35         51     A
PROSEQ              SX4    B4-12                                  001890  CCG35         52     A
PROSEQ              EQ     ISC11                                  001900  CCG35         53     A
PROSEQ                                                                    PROSEQ      1592     A
PROSEQ    ISC9      LX3    R1.INP                                         PROSEQ      1593          I
          -CCG35
PROSEQ    ISC9      MI     X2,ISC9A    IF LCM = G                 001920  CCG35         54          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
          -CCGA044
PROSEQ              LX3    R1.INP                                          001930   CCG35        55          I
          -CCGA044
PROSEQ    *         CHECK IF CA CAN BE CONTAINED IN A 18 BIT FIELD.                  CCGA044      28          I
          -CCGA052
PROSEQ    *         IF NOT THEN WE MUST MATERIALIZE THE CONSTANT IN THE              CCGA044      29          I
          -CCGA052
PROSEQ    *         CONSTANT TABLE.                                                  CCGA044      30          I
          -CCGA052
PROSEQ                                                                               CCGA044      31          I
          -CCGA052
PROSEQ     ISC9     SA2    CHK18                                                    CCGA044      32          I
          -CCGA052
PROSEQ              IX2    X2-X3                                                    CCGA044      33          I
          -CCGA052
PROSEQ              PL     X2,ISC9A    IF CA < CHK18                                CCGA044      34          I
          -CCGA052
PROSEQ    ISC9      BX2    X3                                                       CCGA052      15     A
PROSEQ              AX2    17                                                       CCGA052      16     A
PROSEQ              ZR     X2,ISC9A    IF CA FITS IN 18 BITS                        CCGA052      17     A
PROSEQ              RJ     ILC                                                      CCGA044      35     A
PROSEQ              SX4    B4-8                                                     CCGA044      36     A
PROSEQ              EQ     ISC11                                                    CCGA044      37     A
PROSEQ                                                                              CCGA044      38     A
PROSEQ     ISC9A    LX3    R1.INP                                                   CCGA044      39     A
PROSEQ              BX6    X6+X3                                                    PROSEQ     1594     A
PROSEQ              MX7    0                                                        PROSEQ     1595     A
PROSEQ              SB5    OC.S                                                     PROSEQ     1596     A
PROSEQ                                                                              PROSEQ     1597     A
PROSEQ    ISC10     MX2    0                                                        PROSEQ     1598     A
PROSEQ              RJ     SRI         STORE STT/S                                  PROSEQ     1599     A
PROSEQ              SX4    B4-8        RK = RI(LCM ADDRESS LOAD)                    PROSEQ     1600     A
PROSEQ                                                                              PROSEQ     1601     A
PROSEQ    ISC11     SX5    B4-4        RJ = RI(LCM ADDRESS LOAD)/RI(S/STT)          PROSEQ     1602     A
PROSEQ              LX4    R1.RKP-R1.RJP                                            PROSEQ     1603     A
PROSEQ              BX6    X4+X5                                                    PROSEQ     1604     A
PROSEQ              LX6    R1.RJP                                                   PROSEQ     1605     A
PROSEQ              SA2    LSF                                                      PROSEQ     1606     A
PROSEQ              SX3    B4          RI = NRI                                     PROSEQ     1607     A
PROSEQ              NZ     X2,ISC12    IF EXPANDING LCM LD/ST                       PROSEQ     1608     A
PROSEQ              SA3    A1-2        R1(REF)                                      PROSEQ     1609     A
PROSEQ              MX4    -R1.RIL                                                  PROSEQ     1610     A
PROSEQ              LX3    -R1.RIP                                                  PROSEQ     1611     A
PROSEQ              BX5    -X4*X3      RI(REF)                                      PROSEQ     1612     A
PROSEQ              SX3    X5+B3       RI MOD TO SEQ                                PROSEQ     1613     A
PROSEQ                                                                              PROSEQ     1614     A
PROSEQ    ISC12     LX3    R1.RIP                                                   PROSEQ     1615     A
PROSEQ              BX6    X6+X3                                                    PROSEQ     1616     A
PROSEQ              MX7    0                                                        PROSEQ     1617     A
PROSEQ              SB5    OC.IA                                                    PROSEQ     1618     A
PROSEQ              EQ     ISC0                                                     PROSEQ     1619     A
PROSEQ                                                                      001950   CCG35        56          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

          -CCGA044
PROSEQ     ISC9A    SB5    OC.S                                          001960   CCG35          57          I
          -CCGA044
PROSEQ              MX7    0                                             001970   CCG35          58          I
          -CCGA044
PROSEQ              LX3    R1.INP                                        001980   CCG35          59          I
          -CCGA044
PROSEQ              BX6    X6+X3                                         001990   CCG35          60          I
          -CCGA044
PROSEQ              MX2    0                                             002000   CCG35          61          I
          -CCGA044
PROSEQ              RJ     SRI                                           002010   CCG35          62          I
          -CCGA044
PROSEQ              SX4    B4-8                                          002020   CCG35          63          I
          -CCGA044
PROSEQ              EQ     ISC11                                         002030   CCG35          64          I
          -CCGA044
PROSEQ     ISI      SPACE  4,8         ISI                                       PROSEQ      1620     A
PROSEQ    **        ISI - ISSUE STT INSTRUCTION.                                 PROSEQ      1621     A
PROSEQ    *         ISSUE STT TO MOD USING SEMANTIC INFO OF CURRENT INSTRUCTION  PROSEQ      1622     A
PROSEQ    *         BEING EXPANDED.                                              PROSEQ      1623     A
PROSEQ                                                                           PROSEQ      1624     A
PROSEQ     ISI      ROUTINE                                                      PROSEQ      1625     A
PROSEQ              SA2    A1-B1       IHINFO(REF)                               PROSEQ      1626     A
PROSEQ              SB5    OC.STT                                                PROSEQ      1627     A
PROSEQ              MX1    -IH.RFL                                               PROSEQ      1628          I
          -FCC0912
PROSEQ              LX2    -IH.RFP                                               PROSEQ      1629          I
          -FCC0912
PROSEQ              BX3    -X1*X2      RF(REF)                                   PROSEQ      1630          I
          -FCC0912
PROSEQ              BX7    X2                                                    PROSEQ      1631          I
          -FCC0912
PROSEQ              ZR     X3,ISI1     IF RF .EQ. 0                              PROSEQ      1632          I
          -FCC0912
PROSEQ              SX1    B3          100000B                                   PROSEQ      1633          I
          -FCC0912
PROSEQ              IX7    X1+X2       RF IN SEQ                                 PROSEQ      1634          I
          -FCC0912
PROSEQ              MX6    -IH.RFL                                               FCC0912       33     A
PROSEQ              LX2    -IH.RFP                                               FCC0912       34     A
PROSEQ              BX3    -X6*X2      R = RF[R2]                                FCC0912       35     A
PROSEQ              LX7    X2          R2W = [R2]                                FCC0912       36     A
PROSEQ              ZR     X3,ISI1     IF R = 0                                  FCC0912       37     A
PROSEQ              SX6    B3                                                    FCC0912       38     A
PROSEQ              IX7    X6+X7       RF[R2W] = RF[R2W] + 100000B               FCC0912       39     A
PROSEQ                                                                           PROSEQ      1635     A
PROSEQ     ISI1     LX7    IH.RFP      IHINFO(STT)                               PROSEQ      1636     A
PROSEQ              SX6    B4          RI                                        PROSEQ      1637     A
PROSEQ              SA2    A1-2        R1 WORD                                   FCC1677X       5     A
PROSEQ              LX2    -R1.INP                                               FCC1677X       6     A
PROSEQ              SX3    X2                                                    FCC1677X       7     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ          LX3    R1.INP                                          FCC1677X      8    A
PROSEQ          BX6    X3+X6       TRANSFER UDT ORDINAL TO STT         FCC1677X      9    A
PROSEQ          LX6    R1.RIP                                          PROSEQ     1638         I
         -FCC0912
PROSEQ          MX2    0                                               PROSEQ     1639         I
         -FCC0912
PROSEQ          SX3    B1                                              FCC0912      40    A
PROSEQ          LX3    D.FPP                                           FCC0912      41    A
PROSEQ          BX2    X3*X1       FP = FP[DESCR]                      FCC0912      42    A
PROSEQ          RJ     SRI                                             PROSEQ     1640    A
PROSEQ          EQ     ISI                                             PROSEQ     1641    A
PROSEQ   IAL    SPACE  4,8         IAL                                 PROSEQ     1642    A
PROSEQ   **     IAL - ISSUE ADDRESS LOAD.                              PROSEQ     1643    A
PROSEQ                                                                 PROSEQ     1644    A
PROSEQ   IAL0   RJ     SRI         STORE INSTRUCTION                   PROSEQ     1645         I
         -CCG5
PROSEQ   IAL0   SB5    OC.LD                                           CCG5        932    A
PROSEQ          RJ     SRI                                             CCG5        933    A
PROSEQ                                                                 PROSEQ     1646    A
PROSEQ   IAL    ROUTINE            **ENTRY/EXIT**                      PROSEQ     1647    A
PROSEQ          SX6    B4          RI                                  PROSEQ     1648    A
PROSEQ          SB5    OC.LD                                           PROSEQ     1649    A
PROSEQ          LX6    R1.RIP                                          PROSEQ     1650         I
         -FCC0645
PROSEQ          SX7    B1                                              PROSEQ     1651         I
         -FCC0645
PROSEQ          LX7    IH.RFP      RF = A0                             PROSEQ     1652         I
         -FCC0645
PROSEQ          SX7    B1                                              FCC0645      37    A
PROSEQ          LX7    R1.INP+1    IN[R1W] = 2   */ UDI = 1 ( LOAD ONLY )  FCC0645  38    A
PROSEQ          BX6    X6+X7                                           FCC0645      39    A
PROSEQ          LX7    IH.RFP-R1.INP-1   RF[R2W] = 1  */ A0            FCC0645      40    A
PROSEQ          SA2    A1-B1       IHINFO                              PROSEQ     1653    A
PROSEQ          SX2    X2                                              CCG18       136    A
PROSEQ          MX3    -IH.IHL                                         PROSEQ     1654         I
         -CCG5
PROSEQ          LX2    -IH.IHP                                         PROSEQ     1655         I
         -CCG5
PROSEQ          BX2    -X3*X2      IH(IHINFO)                          PROSEQ     1656         I
         -CCG5
PROSEQ          SX5    X2-2                                            PROSEQ     1657         I
         -CCG5  -CCG18
PROSEQ          SA3    =XO$SYM                                         CCG18       137    A
PROSEQ          LX5    B1,X2                                           CCG18       138    A
PROSEQ          IX2    X5+X2                                           CCG18       139    A
PROSEQ          IX3    X3+X2                                           CCG18       140    A
PROSEQ          SA3    X3+B1                                           CCG18       141    A
PROSEQ          MX2    -WB.FPOL                                        CCG18       142    A
PROSEQ          LX3    -WB.FPOP                                        CCG18       143    A
PROSEQ          BX2    -X2*X3                                          CCG18       144    A
PROSEQ          SX5    X2-1        CA = FPNO[WORDB(IH)] - 1            CCG18       145    A
PROSEQ          SA3    =XO.SYM                                         CCG5        934         I

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

          -CCG18
PROSEQ              SX2    X2                                          CCG5        935          I
          -CCG18
PROSEQ              SB5    X3+B1                                       CCG5        936          I
          -CCG18
PROSEQ              LX3    B1,X2                                       CCG5        937          I
          -CCG18
PROSEQ              IX2    X2+X3                                       CCG5        938          I
          -CCG18
PROSEQ              SA3    B5+X2       WORDB = SYM(3*IH+1)             CCG5        939          I
          -CCG18
PROSEQ              MX2    -WB.FPOL                                    CCG5        940          I
          -CCG18
PROSEQ              LX3    -WB.FPOP                                    CCG5        941          I
          -CCG18
PROSEQ              BX5    -X2*X3      CA = FPO[WORDB]                 CCG5        942          I
          -CCG18
PROSEQ              LX5    IH.CAP                                      PROSEQ     1658     A
PROSEQ              BX7    X7+X5                                       PROSEQ     1659     A
PROSEQ              MX2    0                                           PROSEQ     1660     A
PROSEQ              NZ     X5,IAL0     IF CA = 0                       PROSEQ     1661     A
PROSEQ              MX2    2                                           PROSEQ     1662     A
PROSEQ              LX2    2+D.SZP     INDICATE SIZE = 1               PROSEQ     1663     A
PROSEQ              EQ     IAL0                                        PROSEQ     1664     A
PROSEQ                                                                 PROSEQ     1665     A
PROSEQ    IRA       SPACE  4,8         IRA                             PROSEQ     1666     A
PROSEQ   **         IRA - ISSUE IA FOR RF IF EXISTENT.                 PROSEQ     1667          I
          -FCC0643
PROSEQ   **         IRA - ISSUE *IA* OR *IS* FOR RF IF EXISTANT        FCC0643      52     A
PROSEQ                                                                 PROSEQ     1668     A
PROSEQ    IRA       ROUTINE            **ENTRY/EXIT**                  PROSEQ     1669     A
PROSEQ              SA3    A1-B1       IHINFO(REF)                     PROSEQ     1670     A
PROSEQ              MX2    -IH.RFL                                     PROSEQ     1671     A
PROSEQ              LX3    -IH.RFP                                     PROSEQ     1672     A
PROSEQ              BX4    -X2*X3      RF(REF)                         PROSEQ     1673     A
PROSEQ              ZR     X4,IRA      IF RF.EQ. 0                     PROSEQ     1674          I
          -FCC1918
PROSEQ              SB5    OC.IA                                       PROSEQ     1675          I
          -FCC1918
PROSEQ              ZR     X4,IRA      IF RF[REF] = 0                  FCC1918      13     A
PROSEQ                                                                 FCC1918      14     A
PROSEQ              LX3    IH.RFP+59-IH.SIAP                           FCC1918      15     A
PROSEQ              PL     X3,IRA0     IF ^SIA[REF]  */ NOT INIT STT THAT IS FP&LCM  FCC1918  16  A
PROSEQ                                                                 FCC1918      17     A
PROSEQ              SB5    OC.SA                 */ OUTPUT *SA* TO REMOVE BIT 59  FCC1918  18  A
PROSEQ              SX3    B4-4                                        FCC1918      19     A
PROSEQ              LX3    R1.RJP                                      FCC1918      20     A
PROSEQ              SX6    B4                                          FCC1918      21     A
PROSEQ              MX2    0                                           FCC1918      22     A
PROSEQ              BX6    X3+X6                                       FCC1918      23     A
PROSEQ              MX7    0                                           FCC1918      24     A
PROSEQ              RJ     SRI                                         FCC1918      25     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ                                                                  FCC1918      26     A
PROSEQ    IRA0     SB5    OC.IA                                         FCC1918      27     A
PROSEQ             SA3    O.SEQ                                         PROSEQ     1676     A
PROSEQ             IX1    X3+X4                                         PROSEQ     1677     A
PROSEQ             SA3    X1+2        DI = SEQ(RF) + 2 ;  DIW = [DI]    PROSEQ     1678     A
PROSEQ             LX3    59-D.PIP                                      PROSEQ     1679     A
PROSEQ             MX7    0                                             PROSEQ     1680     A
PROSEQ             PL     X3,IRA1     IF ^PI[DIW]                       PROSEQ     1681          I
          -FCC0643
PROSEQ             PL     X3,IRA3     IF ^PI[DIW]                       FCC0643      53     A
PROSEQ                                                                  PROSEQ     1682     A
PROSEQ    *        GENERATE SA IF RF DEFINED BY DEF TO B-REGISTER.      PROSEQ     1683     A
PROSEQ                                                                  PROSEQ     1684     A
PROSEQ             SA3    X1          R1 = SEQ(RF) ;  R1W = [R1]        PROSEQ     1685     A
PROSEQ             LX3    59-R1.SOP-SO.RTP-1                            PROSEQ     1686     A
PROSEQ             MI     X3,IRA1     IF RT[SO[R1W]] .EQ. 2  */ DEF TO X-REG  PROSEQ  1687  A
PROSEQ             SB5    OC.SA                                         PROSEQ     1688     A
PROSEQ             SA3    =XDIRECT                                      PROSEQ     1689          I
          -CCG1
PROSEQ             SA3    =XCO.LCM                                      CCG1        196          I
          -CCGNC
PROSEQ             SA3    =XHO$LCM                                      CCGNC       744     A
PROSEQ             SA1    A1                                            PROSEQ     1690     A
PROSEQ             ZR     X3,IRA1     IF DIRECT = 0  */ LCM = D         PROSEQ     1691          I
          -CCG1
PROSEQ             ZR     X3,IRA1     IF CO.LCM = 0  */ LCM = D         CCG1        197          I
          -CCGNC
PROSEQ             ZR     X3,IRA1     IF HO$LCM = 0  */ LCM = D         CCGNC       745     A
PROSEQ             LX1    59-D.L2P                                      PROSEQ     1692     A
PROSEQ             PL     X1,IRA1     IF ^L2[DIREF]                     PROSEQ     1693     A
PROSEQ                                                                  PROSEQ     1694     A
PROSEQ    *        LCM = I LEVEL 2 REF.  GENERATE INTERVENING SA FOLLOWED BY IA.  PROSEQ  1695  A
PROSEQ                                                                  PROSEQ     1696     A
PROSEQ             SX5    X4+B3       RJ =  RF + 100000B                PROSEQ     1697     A
PROSEQ             SX4    B4                                            PROSEQ     1698     A
PROSEQ             LX5    R1.RJP                                        PROSEQ     1699     A
PROSEQ             BX6    X4+X5                                         PROSEQ     1700     A
PROSEQ             SX2    B0                                            PROSEQ     1701     A
PROSEQ             RJ     SRI         SRI ( SA, B4, RJ )                PROSEQ     1702     A
PROSEQ             SB5    OC.IA                                         PROSEQ     1703     A
PROSEQ             SX2    B4-8                                          PROSEQ     1704     A
PROSEQ             LX2    R1.RJP-R1.RKP                                 PROSEQ     1705     A
PROSEQ             SX5    B4-4                                          PROSEQ     1706     A
PROSEQ             EQ     IRA2                                          PROSEQ     1707     A
PROSEQ                                                                  PROSEQ     1708     A
PROSEQ    IRA1     SX2    B4-4                                          PROSEQ     1709     A
PROSEQ             LX2    R1.RJP-R1.RKP                                 PROSEQ     1710     A
PROSEQ             SX5    X4+B3       RF MOD TO SEQ                     PROSEQ     1711     A
PROSEQ                                                                  PROSEQ     1712     A
PROSEQ    IRA2     BX5    X2+X5                                         PROSEQ     1713     A
PROSEQ             LX5    R1.RKP                                        PROSEQ     1714     A
PROSEQ             SX4    B4          RI                                PROSEQ     1715     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            LX4    R1.RIP                                          PROSEQ     1716    A
PROSEQ            BX6    X4+X5       R1 = TYPE1(RI,RI-4,RF)              PROSEQ     1717    A
PROSEQ            SX2    B0                                              PROSEQ     1718    A
PROSEQ            RJ     SRI         STORE IA                            PROSEQ     1719          I
          -FCC0643
PROSEQ            RJ     SRI         SRI(  IA/IS RI,RI-4,RF )            FCC0643      54    A
PROSEQ            SA3    A6                                              FCC1239       5    A
PROSEQ            SX2    B1                                              FCC1239       6    A
PROSEQ            LX2    D.USIP                                          FCC1239       7    A
PROSEQ            BX6    X2+X3                                           FCC1239       8    A
PROSEQ            SA6    A3          USI[D] = 1                          FCC1239       9    A
PROSEQ            EQ     IRA                                             PROSEQ     1720    A
PROSEQ                                                                   FCC0643      55    A
PROSEQ    IRA3    SA3    X1          R1W = SEQ(RF)                       FCC0643      56    A
PROSEQ            BX4    X3                                              FCC0643      57    A
PROSEQ            AX3    R1.OCP                                          FCC0643      58    A
PROSEQ            SX6    X3-OC.XMTC-2000B                                FCC0643      59    A
PROSEQ            NZ     X6,IRA1     IF OC[R1W] " OC.XMTC                FCC0643      60    A
PROSEQ                                                                   FCC0643      61    A
PROSEQ    *       A(C-I) , GENERATE *IS* AND SET RF.MV BIT IN *RF*       FCC0643      62    A
PROSEQ                                                                   FCC0643      63    A
PROSEQ            LX4    -R1.RJP                                         FCC0643      64    A
PROSEQ            MX2    -R1.RIL                                         FCC0643      65    A
PROSEQ            SB5    OC.IS                                           FCC0643      66    A
PROSEQ            BX4    -X2*X4      RF = RJ[R1W]  */ XMTC PRED          FCC0643      67    A
PROSEQ            SA5    R2W                                             FCC0643      68    A
PROSEQ            LX2    IH.RFP                                          FCC0643      69    A
PROSEQ            SX6    X4+RF.MV                                        FCC0643      70    A
PROSEQ            BX5    X2*X5                                           FCC0643      71    A
PROSEQ            LX6    IH.RFP                                          FCC0643      72    A
PROSEQ            BX6    X5+X6       RF[R2W] = RF + RF.MV                FCC0643      73    A
PROSEQ            SA6    A5                                              FCC0643      74    A
PROSEQ            EQ     IRA1                                            FCC0643      75    A
PROSEQ    ISX     SPACE  4,8         ISX                                 PROSEQ     1721    A
PROSEQ    **      ISX - ISSUE STT OR XMT.                                PROSEQ     1722    A
PROSEQ    *       GENERATE CODE FOR SET TO FORMAL PARAMETER ADDRESS FOLLOWING  PROSEQ 1723  A
PROSEQ    *       ADDRESS LOAD.                                          PROSEQ     1724    A
PROSEQ                                                                   PROSEQ     1725    A
PROSEQ    ISX0    MX2    0                                               PROSEQ     1726    A
PROSEQ            RJ     SRI         STORE INSTRUCTION                   PROSEQ     1727    A
PROSEQ                                                                   PROSEQ     1728    A
PROSEQ    ISX     ROUTINE            **ENTRY/EXIT**                      PROSEQ     1729    A
PROSEQ            SA2    A1-B1       IHINFO(REF)                         PROSEQ     1730    A
PROSEQ            MX1    -IH.CAL                                         PROSEQ     1731    A
PROSEQ            LX2    -IH.CAP                                         PROSEQ     1732    A
PROSEQ            BX3    -X1*X2      CA(REF)                             PROSEQ     1733    A
PROSEQ            SX2    B4-4        RP = RI(ADDRESS LOAD)               PROSEQ     1734    A
PROSEQ            SX6    B4          RI = NRI                            PROSEQ     1735    A
PROSEQ            SA5    LSF                                             PROSEQ     1736    A
PROSEQ            NZ     X5,ISX1     IF EXPANDING LD OR ST               PROSEQ     1737    A
PROSEQ            SA4    A2-B1       R1(REF)                             PROSEQ     1738    A
PROSEQ            MX1    -R1.RIL                                         PROSEQ     1739    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            LX4    -R1.RIP                                          PROSEQ     1740    A
PROSEQ            BX1    -X1*X4      RI(REF)                              PROSEQ     1741    A
PROSEQ            SX6    X1+B3       RI = RI(REF) + 100000B               PROSEQ     1742    A
PROSEQ                                                                    PROSEQ     1743    A
PROSEQ    ISX1    LX6    R1.RIP                                           PROSEQ     1744    A
PROSEQ            ZR     X3,ISX2     IF CA .EQ. 0                         PROSEQ     1745    A
PROSEQ            LX2    IH.RFP      RF = RP                              PROSEQ     1746    A
PROSEQ            SB5    OC.STT                                           PROSEQ     1747    A
PROSEQ            LX3    IH.CAP                                           PROSEQ     1748    A
PROSEQ            BX7    X2+X3                                            PROSEQ     1749    A
PROSEQ            NZ     X5,ISX0     IF EXPANDING LD OR ST                PROSEQ     1750    A
PROSEQ            SA1    A1          DI = D(I)                            PROSEQ     1751    A
PROSEQ            LX1    59-D.L2P                                         PROSEQ     1752    A
PROSEQ            PL     X1,ISX0     IF ^L2[DI]                           PROSEQ     1753    A
PROSEQ            SX6    B4          RI = NRI                             PROSEQ     1754    A
PROSEQ            LX6    R1.RIP                                           PROSEQ     1755    A
PROSEQ            MX2    0                                                PROSEQ     1756    A
PROSEQ            RJ     SRI         STORE INSTRUCTION                    PROSEQ     1757    A
PROSEQ            RJ     IMO         ISSUE MASK-OR BIT 59                 PROSEQ     1758    A
PROSEQ            EQ     ISX                                              PROSEQ     1759    A
PROSEQ                                                                    PROSEQ     1760    A
PROSEQ    ISX2    NZ     X5,ISX      IF EXPANDING LOAD OR STORE           PROSEQ     1761    A
PROSEQ            RJ     IXI         ISSUE A *SA/XMT*                     PROSEQ     1762    A
PROSEQ            EQ     ISX                                              PROSEQ     1763    A
PROSEQ    IXI     SPACE  3           IXI                                  PROSEQ     1764          I
          -CCG5
PROSEQ    IXI     SPACE  3,14                                             CCG5        943    A
PROSEQ    **      IXI - ISSUE *XMT/SA* INSTRUCTION                        PROSEQ     1765    A
PROSEQ                                                                    PROSEQ     1766    A
PROSEQ    IXI     ROUTINE                                                 PROSEQ     1767    A
PROSEQ            SA1    A1                                               PROSEQ     1768    A
PROSEQ            SB5    OC.XMT                                           PROSEQ     1769    A
PROSEQ            SA3    HO$LCM                                  002050   CCG35        65          I
          -CCG0004
PROSEQ            MI     X3,IXI1     IF LCM = G                  002060   CCG35        66          I
          -CCG0004
PROSEQ            LX1    59-D.PRSP                                        PROSEQ     1770    A
PROSEQ            PL     X1,IXI1     IF ^PRS[DI]                          PROSEQ     1771    A
PROSEQ            SA3    A1+2        R1RS = DI + 2                        FCC1596X      1    A
PROSEQ            LX3    58-R1.SOP-SO.RTP                                 FCC1596X      2    A
PROSEQ            MI     X3,IXI1     IF SORT [R1RS] = 2  */ RS TO AN X-REG FCC1596X     3    A
PROSEQ            SB5    OC.SA                                            PROSEQ     1772    A
PROSEQ    IXI1    LX2    R1.RJP                                           PROSEQ     1773    A
PROSEQ            SX7    B0                                               PROSEQ     1774    A
PROSEQ            BX6    X2+X6                                            PROSEQ     1775    A
PROSEQ            MX2    0                                                PROSEQ     1776    A
PROSEQ            RJ     SRI                                              PROSEQ     1777    A
PROSEQ            EQ     IXI                                              PROSEQ     1778    A
PROSEQ    IPR     SPACE  4,8         IPR                                  PROSEQ     1779    A
PROSEQ    **      IPR - ISSUE PARAMETER REFERENCE.                        PROSEQ     1780    A
PROSEQ                                                                    PROSEQ     1781    A
PROSEQ    IPR0    RJ     SRI         STORE INSTRUCTION                    PROSEQ     1782    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ                                                                  PROSEQ      1783     A
PROSEQ    IPR      ROUTINE            **ENTRY/EXIT**                    PROSEQ      1784     A
PROSEQ             SA2    A1-2        R1(REF)                           PROSEQ      1785     A
PROSEQ             LX2    -R1.RIP                                       PROSEQ      1786     A
PROSEQ             SA3    LSF                                           PROSEQ      1787     A
PROSEQ             SB5    OC.PLD+2+X3 OC.PLD/PST                        PROSEQ      1788     A
PROSEQ             MX3    -R1.RIL                                       PROSEQ      1789     A
PROSEQ             BX4    -X3*X2      RI(REF)                           PROSEQ      1790     A
PROSEQ             SX2    X4+B3       RI(PLD/PST) = RI(REF) + 100000B   PROSEQ      1791     A
PROSEQ             LX2    R1.RIP                                        PROSEQ      1792     A
PROSEQ             SX4    B4-4                                          PROSEQ      1793     A
PROSEQ             LX4    R1.RJP                                        PROSEQ      1794     A
PROSEQ             BX6    X2+X4                                         PROSEQ      1795     A
PROSEQ             SA5    A1-B1       IHINFO(REF)                       PROSEQ      1796          I
          -FCC0643
PROSEQ             SA5    R2W         IHINFO = R2W                      FCC0643       76     A
PROSEQ             SX1    B3          OFFSET FOR MPB                    PROSEQ      1797     A
PROSEQ             LX1    IH.RFP                                        PROSEQ      1798     A
PROSEQ             IX7    X1+X5       IHINFO(PLD/PST)                   PROSEQ      1799     A
PROSEQ             MX2    0                                             PROSEQ      1800     A
PROSEQ             LX5    -IH.CAP                                       PROSEQ      1801     A
PROSEQ             SX4    X5                                            PROSEQ      1802     A
PROSEQ             NZ     X4,IPR0     IF CA .NE. 0                      PROSEQ      1803     A
PROSEQ             MX2    2                                             PROSEQ      1804     A
PROSEQ             LX2    2+D.SZP     INDICATE TOGGLE SIZE FIELD        PROSEQ      1805     A
PROSEQ             EQ     IPR0                                          PROSEQ      1806     A
PROSEQ    IDR      SPACE  4,8         IDR                               PROSEQ      1807     A
PROSEQ   **        IDR - ISSUE DIRECT LCM REFERENCE.                    PROSEQ      1808     A
PROSEQ                                                                  PROSEQ      1809     A
PROSEQ    #DAL     IFNE   .DAL,0      IF LEVEL 2 ALLOWED                PROSEQ      1810     A
PROSEQ                                                                  PROSEQ      1811     A
PROSEQ    IDR      ROUTINE            **ENTRY/EXIT**                    PROSEQ      1812     A
PROSEQ             SX5    B3          100000B FOR MPB                   PROSEQ      1813     A
PROSEQ             SA2    A1-B1       IHINFO                            PROSEQ      1814          I
          -FCC0643
PROSEQ             SA2    R2W         IHINFO = R2W                      FCC0643       77     A
PROSEQ             SA4    LSF                                           PROSEQ      1815     A
PROSEQ             SB5    OC.DRL+2+X4 OC.DRL/DWL                        PROSEQ      1816     A
PROSEQ             LX5    IH.RFP                                        PROSEQ      1817     A
PROSEQ             IX7    X2+X5       RF MOD TO SEQ                     PROSEQ      1818     A
PROSEQ             MX3    -R1.RIL                                       PROSEQ      1819     A
PROSEQ             SA2    A2-B1       R1 WORD                           PROSEQ      1820          I
          -FCC0643
PROSEQ             SA2    A1-2        R1I = DI - 2                      FCC0643       78     A
PROSEQ             SX5    IH.LCM                                        PROSEQ      1821     A
PROSEQ             LX5    IH.IHP                                        PROSEQ      1822     A
PROSEQ             IX7    X7+X5                                         PROSEQ      1823     A
PROSEQ             LX2    -R1.RIP                                       PROSEQ      1824     A
PROSEQ             BX4    -X3*X2      RI(REF)                           PROSEQ      1825     A
PROSEQ             SX5    B4-4        RJ = RI(PRED)                     PROSEQ      1826     A
PROSEQ             SX4    X4+B3       RI MOD TO SEQ                     PROSEQ      1827     A
PROSEQ             LX4    R1.RIP                                        PROSEQ      1828     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ          LX5    R1.RJP                                          PROSEQ     1829    A
PROSEQ          BX6    X4+X5                                           PROSEQ     1830    A
PROSEQ          MX2    0                                               PROSEQ     1831          I
         -CCG35
PROSEQ          SA2    LEV0                                   002080   CCG35        67    A
PROSEQ          RJ     SRI                                             PROSEQ     1832    A
PROSEQ          EQ     IDR                                             PROSEQ     1833    A
PROSEQ                                                                 PROSEQ     1834    A
PROSEQ    #DAL  ENDIF                                                  PROSEQ     1835    A
PROSEQ    IMO   SPACE  4,8         IMO                                 PROSEQ     1836    A
PROSEQ    **    IMO - ISSUE MASK-OR BIT 59.                            PROSEQ     1837    A
PROSEQ    *     IMO GENERATES THE CODE NECESSARY TO INSURE THAT BIT 59 IS PROSEQ  1838    A
PROSEQ    *     SET FOR LCM APLIST ENTRIES AND LOCF/S.  THIS CODE IS NEVER PROSEQ  1839    A
PROSEQ    *     REQUIRED IN LCM=I MODE SINCE 60-BIT ARITHMETIC ALWAYS  PROSEQ     1840    A
PROSEQ    *     PRESERVES BIT 59.  WE ALSO DO NOT NEED THIS CODE IN DIRECT PROSEQ  1841    A
PROSEQ    *     MODE IF NO SET HAS BEEN PERFORMED ON THE ADDRESS QUANTITY, PROSEQ  1842    A
PROSEQ    *     FOR EXAMPLE LOCF(FP).                                  PROSEQ     1843    A
PROSEQ                                                                 PROSEQ     1844    A
PROSEQ    IMO   ROUTINE                                                PROSEQ     1845    A
PROSEQ          SB5    OC.FMA                                          PROSEQ     1846    A
PROSEQ          SX5    B1                                              PROSEQ     1847    A
PROSEQ          LX5    R1.INP                                          PROSEQ     1848    A
PROSEQ          SX4    B4        RI = NRI                              PROSEQ     1849    A
PROSEQ          LX4    R1.RIP                                          PROSEQ     1850    A
PROSEQ          BX6    X4+X5                                           PROSEQ     1851    A
PROSEQ          MX7    0                                               PROSEQ     1852    A
PROSEQ          SX2    B0                                              PROSEQ     1853    A
PROSEQ          RJ     SRI       STORE FMA(NRI,1)                      PROSEQ     1854    A
PROSEQ                                                                 PROSEQ     1855    A
PROSEQ          SA4    A1-2      R = R1(I)                             PROSEQ     1856    A
PROSEQ          MX1    -R1.RIL                                         PROSEQ     1857    A
PROSEQ          LX4    -R1.RIP                                         PROSEQ     1858    A
PROSEQ          SB5    OC.OR                                           PROSEQ     1859    A
PROSEQ          BX5    -X1*X4    RIR = R1F(R)                          PROSEQ     1860    A
PROSEQ          SX6    X5+B3     R1N = RIR + 100000B                   PROSEQ     1861    A
PROSEQ          LX6    R1.RIP                                          PROSEQ     1862    A
PROSEQ          SX3    B4-4      RJ = NRI - 4                          PROSEQ     1863    A
PROSEQ          SX4    X3-4      RK = NRI - 8                          PROSEQ     1864    A
PROSEQ          LX3    R1.RJP                                          PROSEQ     1865    A
PROSEQ          BX6    X6+X3                                           PROSEQ     1866    A
PROSEQ          LX4    R1.RKP                                          PROSEQ     1867    A
PROSEQ          BX6    X6+X4                                           PROSEQ     1868    A
PROSEQ          MX7    0                                               PROSEQ     1869    A
PROSEQ          SX2    B0                                              PROSEQ     1870    A
PROSEQ          RJ     SRI       STORE OR(RIN,NRI-4,NRI-8)             PROSEQ     1871    A
PROSEQ          EQ     IMO                                             PROSEQ     1872    A
PROSEQ                                                                 CCGA044      40          I
         -CCGA052
PROSEQ    ILC   SPACE  4,10                                            CCGA052      18    A
PROSEQ    **    ILC - ISSUE A LOAD OF A CONSTANT                       CCGA044      41    A
PROSEQ    *     ENTRY  X3 = VALUE OF CONSTANT TO BE LOADED             CCGA044      42    A
PROSEQ    *            B4 = CURRENT R NUMBER                           CCGA044      43    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ   *                                                                  CCGA044    44    A
PROSEQ   *        EXIT   A0,A1,B1,B2,B3,X0,X4 PRESERVED.                    CCGA044    45    A
PROSEQ   *               B4 AS LAST UPDATED BY SRI.                         CCGA044    46    A
PROSEQ   *               EXPANSION FILED IN MOD, WITH A7 UPDATED            CCGA044    47    A
PROSEQ   *                                                                  CCGA044    48    A
PROSEQ                                                                      CCGA044    49    A
PROSEQ    ILC     ROUTINE            ** ENTRY/EXIT **                       CCGA044    50    A
PROSEQ            SA2    A7                                                 CCGA044    51          I
         -CCG0088
PROSEQ            SA2    O.MOD                                              CCG0088    14    A
PROSEQ            SX1    A0                                                 CCGA044    52    A
PROSEQ            SX5    A1                                                 CCGA044    53    A
PROSEQ            BX6    X2                                                 CCGA044    54          I
         -CCG0088
PROSEQ            IX1    X1-X2                                              CCG0088    15    A
PROSEQ            LX1    18                                                 CCGA044    55    A
PROSEQ            SA6    SAVREG      SAVE CONTENTS OF A7                    CCGA044    56          I
         -CCG0088
PROSEQ            SX7    A7                                                 CCG0088    16    A
PROSEQ            IX6    X7-X2                                              CCG0088    17    A
PROSEQ            SA6    SAVREG      SAVE MOD OFFSET                        CCG0088    18    A
PROSEQ            BX7    X1+X5                                              CCGA044    57    A
PROSEQ            SX2    A7                                                 CCGA044    58    A
PROSEQ            LX7    18                                                 CCGA044    59    A
PROSEQ            BX6    X2+X7                                              CCGA044    60    A
PROSEQ            MX1    -18                                                CCGA044    61    A
PROSEQ            SA6    A6+B1       SAVE A0,A1,A7                          CCGA044    62    A
PROSEQ            SX2    B2                                                 CCGA044    63    A
PROSEQ            SX5    B3                                                 CCGA044    64    A
PROSEQ            BX7    -X1*X2                                             CCGA044    65    A
PROSEQ            SX6    B4                                                 CCGA044    66    A
PROSEQ            LX7    18                                                 CCGA044    67    A
PROSEQ            BX2    -X1*X5                                             CCGA044    68    A
PROSEQ            BX7    X2+X7                                              CCGA044    69    A
PROSEQ            BX6    -X1*X6                                             CCGA044    70    A
PROSEQ            LX7    18                                                 CCGA044    71    A
PROSEQ            BX7    X1*X7                                              CCGA044    72    A
PROSEQ            BX7    X6+X7                                              CCGA044    73    A
PROSEQ            SA7    A6+B1       SAVE B2,B3,B4                          CCGA044    74    A
PROSEQ            BX6    X0                                                 CCGA044    75    A
PROSEQ            BX7    X4                                                 CCGA044    76    A
PROSEQ            SA6    A7+B1       SAVE X0                                CCGA044    77    A
PROSEQ            SA7    A6+B1       SAVE X4                                CCGA044    78    A
PROSEQ                                                                      CCGA044    79    A
PROSEQ   *        ALL NECESSARY REGISTERS HAVE BEEN SAVED.                  CCGA044    80    A
PROSEQ   *        NOW RESERVE A SPOT IN THE CONSTANT VALUE TABLE (CVL)      CCGA044    81    A
PROSEQ   *        FOR THE CA VALUE AND EMIT A LDC OF THE CONSTANT.          CCGA044    82    A
PROSEQ                                                                      CCGA044    83    A
PROSEQ            BX1    X3                                                 CCGA044    84    A
PROSEQ            CALL   CG$SCT                                             CCGA044    85    A
PROSEQ                                                                      CCGA044    86    A
PROSEQ   *        MUST RESTORE B4 AND A7 BEFORE SRI CAN BE CALLED.          CCGA044    87    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ                                                                  CCGA044     88    A
PROSEQ            SA1    SAVREG                                         CCGA044     89    A
PROSEQ            SA4    A1+B1                                          CCGA044     90    A
PROSEQ            SA5    A4+B1                                          CCGA044     91    A
PROSEQ            BX7    X1                                             CCGA044     92          I
          -CCG0088
PROSEQ            SB4    X5          RESTORE B4                         CCGA044     93          I
          -CCG0088
PROSEQ            SA7    X4          RESTORE A7 AND ITS CONTENTS        CCGA044     94          I
          -CCG0088
PROSEQ            SA2    O.MOD                                          CCG0088     19    A
PROSEQ            IX1    X1+X2                                          CCG0088     20    A
PROSEQ            LX2    36                                             CCG0088     21    A
PROSEQ            IX4    X2+X4       RELOCATE SAVED A0                  CCG0088     22    A
PROSEQ            MX7    0                                              CCG0088     23    A
PROSEQ            SA7    X1          RESTORE X7                         CCG0088     24    A
PROSEQ            SB4    X5          RESTORE B4                         CCG0088     25    A
PROSEQ                                                                  CCGA044     95    A
PROSEQ   *        ISSUE LDC OF ORDINAL RETURNED FROM CG$SCT             CCGA044     96    A
PROSEQ                                                                  CCGA044     97    A
PROSEQ            SA2    =XS=CON     IH = CON.                          CCGA044     98    A
PROSEQ            SB5    OC.LDC                                         CCGA044     99    A
PROSEQ            LX6    IH.CAP      CA = CON. ORDINAL                  CCGA044    100    A
PROSEQ            BX7    X2+X6                                          CCGA044    101    A
PROSEQ            MX2    0                                              CCGA044    102    A
PROSEQ            SX6    B4                                             CCGA044    103    A
PROSEQ            RJ     SRI                                            CCGA044    104    A
PROSEQ                                                                  CCGA044    105    A
PROSEQ   *        RESTORE THE REST OF THE REGISTERS                     CCGA044    106    A
PROSEQ                                                                  CCGA044    107    A
PROSEQ            SA2    A5+B1                                          CCGA044    108    A
PROSEQ            AX4    18                                             CCGA044    109    A
PROSEQ            BX0    X2                                             CCGA044    110    A
PROSEQ            AX5    18                                             CCGA044    111    A
PROSEQ            SA1    X4                                             CCGA044    112    A
PROSEQ            SB3    X5                                             CCGA044    113    A
PROSEQ            AX4    18                                             CCGA044    114    A
PROSEQ            SA0    X4                                             CCGA044    115    A
PROSEQ            AX5    18                                             CCGA044    116    A
PROSEQ            SB2    X5                                             CCGA044    117    A
PROSEQ            SA4    A2+B1                                          CCGA044    118    A
PROSEQ            EQ     ILC                                            CCGA044    119    A
PROSEQ                                                                  CCGA044    120    A
PROSEQ                                                                  CCGA044    121    A
PROSEQ    SAVREG  BSS    5                                              CCGA044    122    A
PROSEQ                                                                  CCGA044    123    A
PROSEQ    SRI     SPACE  4,8         SRI                               PROSEQ     1873    A
PROSEQ   **       SRI - STORE R-LIST INSTRUCTION.                      PROSEQ     1874    A
PROSEQ   *        ENTRY  B5 = OP-CODE                                  PROSEQ     1875    A
PROSEQ   *               X6 = RLIST1 WORD                              PROSEQ     1876    A
PROSEQ   *               X7 = RLIST2 WORD                              PROSEQ     1877    A
PROSEQ   *               X2 = 0 OR 3S D.SZP IF SIZE FIELD IS TO BE RESET  PROSEQ  1878    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ                                                                                  PROSEQ      1879     A
PROSEQ    SRI      ROUTINE            **ENTRY/EXIT**                                    PROSEQ      1880     A
PROSEQ             PX6    B5,X6                                                        PROSEQ      1881     A
PROSEQ             SB4    B4+4        RI = RI + 4                                      PROSEQ      1882     A
PROSEQ             SA6    A7+B1                                                        PROSEQ      1883     A
PROSEQ             SA7    A6+B1                                                        PROSEQ      1884     A
PROSEQ             SA3    =XF.RDT+B5                                                   PROSEQ      1885     A
PROSEQ             BX6    X2-X3       TOGGLE SIZE FIELD IF REQUESTED                   PROSEQ      1886     A
PROSEQ             SA6    A7+B1                                                        PROSEQ      1887     A
PROSEQ             MX7    0                                                            PROSEQ      1888     A
PROSEQ             SA7    A6+B1                                                        PROSEQ      1889     A
PROSEQ             EQ     SRI                                                          PROSEQ      1890     A
PROSEQ    TRE      SPACE  4,8         TRE                                              PROSEQ      1891     A
PROSEQ   **        TRE - TERMINATE REF EXPANSION.                                      PROSEQ      1892     A
PROSEQ                                                                                 PROSEQ      1893     A
PROSEQ    TRE      ROUTINE            **ENTRY/EXIT**                                    PROSEQ      1894     A
PROSEQ             SB7    A0          IMA                                              PROSEQ      1895     A
PROSEQ             SX4    A7-B7                                                        PROSEQ      1896     A
PROSEQ             ZR     X4,TRE      IF NO EXPANSION                                  PROSEQ      1897     A
PROSEQ                                                                                 PROSEQ      1898     A
PROSEQ             SA3    O.SEQ                                                        PROSEQ      1899     A
PROSEQ             SB6    X3+2                                                         PROSEQ      1900     A
PROSEQ             SX5    A1-B6       R1 INDEX IN TXT                                  PROSEQ      1901     A
PROSEQ             ERRMI  ML.NIP-2                                                     PROSEQ      1902     A
PROSEQ             LX4    ML.NIP-2    NI = N.WORDS EXPANSION/4                         PROSEQ      1903     A
PROSEQ             LX5    ML.IIP                                                       PROSEQ      1904     A
PROSEQ             SA3    O.MOD                                                        PROSEQ      1905     A
PROSEQ             BX1    X4+X5                                                        PROSEQ      1906     A
PROSEQ             SB6    X3-1                                                         PROSEQ      1907     A
PROSEQ             SX6    B7-B6       INDEX IN MOD                                     PROSEQ      1908     A
PROSEQ             LX6    ML.MTIP                                                      PROSEQ      1909     A
PROSEQ             MX7    1                                                            PROSEQ      1910     A
PROSEQ             LX7    1+ML.DELP                                                    PROSEQ      1911     A
PROSEQ             BX3    X1+X6                                                        PROSEQ      1912     A
PROSEQ             SA4    L.MLT                                                        PROSEQ      1913     A
PROSEQ             BX6    X7+X3                                                        PROSEQ      1914     A
PROSEQ             SA3    O.MLT                                                        PROSEQ      1915     A
PROSEQ             IX5    X3+X4                                                        PROSEQ      1916     A
PROSEQ             SA6    X5          MLT=                                             PROSEQ      1917     A
PROSEQ             SX6    X4+B1                                                        PROSEQ      1918     A
PROSEQ             SA6    A4          L.MLT = L.MLT + 1                                PROSEQ      1919     A
PROSEQ             EQ     TRE                                                          PROSEQ      1920     A
PROSEQ                                                                                 CCG13         87     A
PROSEQ    .AET     ELSE                                                                CCG13         88     A
PROSEQ                                                                                 CCG13         89     A
PROSEQ    ESRA     ENTRY. ,#                                                           CCG13         90     A
PROSEQ    ESR      ENTRY. **,#                                                         CCG13         91     A
PROSEQ             EQ     *+1S17                                                       CCG13         92     A
PROSEQ                                                                                 CCG5         944     A
PROSEQ    .AET     ENDIF                                                               CCG5         945     A
PROSEQ    OPR      TITLE  F.RDT - RLIST INSTRUCTION DESCRIPTOR TABLE                   PROSEQ      1921         I
         -CCG5

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            LIST   -X,-R,-F                                                PROSEQ     1922        I
          -CCG5
PROSEQ            ECHO   2,A=(I,II,III,IV,BSSZ),B=(0,1,2,3,1)                    PROSEQ     1923        I
          -CCG5
PROSEQ            NOREF  A                                                       PROSEQ     1924        I
          -CCG5
PROSEQ    A       EQU    B                                                       PROSEQ     1925        I
          -CCG5
PROSEQ                                                                           PROSEQ     1926        I
          -CCG5
PROSEQ    TESTM   IFNE   TEST,0                                                  PROSEQ     1927        I
          -CCG5
PROSEQ            MACRO  OPR,NAME,SS                                             PROSEQ     1928        I
          -CCG5
PROSEQ    TY      SET    SS                                                      PROSEQ     1929        I
          -CCG5
PROSEQ            CON    TY+1+4L_NAME                                            PROSEQ     1930        I
          -CCG5
PROSEQ            ENDM                                                           PROSEQ     1931        I
          -CCG5
PROSEQ                                                                           PROSEQ     1932        I
          -CCG5
PROSEQ            USE    /OPRS/                                                  PROSEQ     1933        I
          -CCG5
PROSEQ    *CALL   OPRDEFS                                                        PROSEQ     1934        I
          -CCG5
PROSEQ            USE    *                                                       PROSEQ     1935        I
          -CCG5
PROSEQ    TESTM   ENDIF                                                          PROSEQ     1936        I
          -CCG5
PROSEQ            LIST   -R,X,F                                                  PROSEQ     1937        I
          -CCG5
PROSEQ    OPR     SPACE  4                                                       PROSEQ     1938        I
          -CCG5
PROSEQ    **      OPR - DEFINE *OPR* MACRO TO FORM RLIST DESCRIPTOR TABLE        PROSEQ     1939        I
          -CCG5
PROSEQ                                                                           PROSEQ     1940        I
          -CCG5
PROSEQ    *       DEFINE VALUES FOR USE IN *OPR* MACRO CALLS                     PROSEQ     1941        I
          -CCG5
PROSEQ                                                                           PROSEQ     1942        I
          -CCG5
PROSEQ    MC      SET    34-15                                                   PROSEQ     1943        I
          -CCG5
PROSEQ            IFEQ   .CPU,76,1                                               PROSEQ     1944        I
          -CCG5
PROSEQ    MC      SET    37-15                                                   PROSEQ     1945        I
          -CCG5
PROSEQ                                                                           PROSEQ     1946        I
          -CCG5
PROSEQ            ECHO   2,X=(COMM,USI,),V=(1,1,0)                               PROSEQ     1947        I
          -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            NOREF  .X                                                    PROSEQ     1948        I
          -CCG5
PROSEQ     .X     =      V                                                     PROSEQ     1949        I
          -CCG5
PROSEQ                                                                         PROSEQ     1950        I
          -CCG5
PROSEQ    *       SYMBOL DEFINED IF INSTRUCTION DOESN"T DEFINE AN RI           PROSEQ     1951        I
          -CCG5
PROSEQ                                                                         PROSEQ     1952        I
          -CCG5
PROSEQ            ECHO   2,N=(JUMP,UJUMP,STORE,STORE2,REGST,ECJUMP)            PROSEQ     1953        I
          -CCG5
PROSEQ            NOREF  Y.N                                                   PROSEQ     1954        I
          -CCG5
PROSEQ     Y.N    EQU    1                                                     PROSEQ     1955        I
          -CCG5
PROSEQ                                                                         PROSEQ     1956        I
          -CCG5
PROSEQ            ECHO   2,N=(LOAD,STORE,JUMP,UJUMP,LOAD2,STORE2,ECJUMP),_______ PROSEQ     1957        I
          -CCG5
PROSEQ    ,V=(4,2,1,1,14B,12B,20B)                                             PROSEQ     1958        I
          -CCG5
PROSEQ            NOREF  Z.N                                                   PROSEQ     1959        I
          -CCG5
PROSEQ     Z.N    EQU    V                                                     PROSEQ     1960        I
          -CCG5
PROSEQ                                                                         PROSEQ     1961        I
          -CCG5
PROSEQ    *       BITS SET FOR XMT , CON SHIFTS AND PSEUDO INSTRUCTION FIELDS  PROSEQ     1962        I
          -CCG5
PROSEQ                                                                         PROSEQ     1963        I
          -CCG5
PROSEQ            ECHO   2,N=(SXT,XMT,KLS,KRS,EOQ,BOS,EOS,DAR,RS,DEF,LAB,ENT),__ PROSEQ     1964        I
          -CCG5
PROSEQ    ,V=(4,4,2,2,1,1,1,1,1,1,1,1)                                         PROSEQ     1965        I
          -CCG5
PROSEQ            NOREF  S.N                                                   PROSEQ     1966        I
          -CCG5
PROSEQ     S.N    EQU    V                                                     PROSEQ     1967        I
          -CCG5
PROSEQ                                                                         PROSEQ     1968        I
          -CCG5
PROSEQ            MACRO  OPR,NAM,SS                                            PROSEQ     1969        I
          -CCG5
PROSEQ            ECHO   1,X=(BD,LSJ,XSP)                                      PROSEQ     1970        I
          -CCG5
PROSEQ     X      SET    0                                                     PROSEQ     1971        I
          -CCG5
PROSEQ     TYPE   SET    SS                                                    PROSEQ     1972        I
          -CCG5
PROSEQ     S      GETARG 23,4,(SS)                                             PROSEQ     1973        I
          -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ    CO       SET    ."S"                                                    PROSEQ    1974      I
          -CCG5
PROSEQ    *                          UNSAFE INSTRUCTION                          PROSEQ    1975      I
          -CCG5
PROSEQ    S        GETARG 29,3,(SS)                                               PROSEQ    1976      I
          -CCG5
PROSEQ    CO       SET    CO+4*."S"                                               PROSEQ    1977      I
          -CCG5
PROSEQ    *                       FUNCTION UNIT AND TIMES                        PROSEQ    1978      I
          -CCG5
PROSEQ    S        MICRO  MC,2,$SS$                                               PROSEQ    1979      I
          -CCG5
PROSEQ    FT       SET    "S" 0                                                   PROSEQ    1980      I
          -CCG5
PROSEQ    JFT      SET    FT/FT                                                   PROSEQ    1981      I
          -CCG5
PROSEQ    S        GETARG 40,6,(SS)                                               PROSEQ    1982      I
          -CCG5
PROSEQ    FU1      SET    0                                                       PROSEQ    1983      I
          -CCG5
PROSEQ             IFC    NE,//"S"/,1                                             PROSEQ    1984      I
          -CCG5
PROSEQ    FU1      SET    /FUNIT/"S"                                              PROSEQ    1985      I
          -CCG5
PROSEQ    FU2      SET    FU1                                                     PROSEQ    1986      I
          -CCG5
PROSEQ    *                       SECOND FUNCTIONAL UNIT ( 6600 )                PROSEQ    1987      I
          -CCG5
PROSEQ    #MD      IFEQ   .CPU,74                                   ?6600         PROSEQ    1988      I
          -CCG5
PROSEQ    S        GETARG 47,5,(SS)                                               PROSEQ    1989      I
          -CCG5
PROSEQ             IFC    NE,/"S"//,3                                             PROSEQ    1990      I
          -CCG5
PROSEQ    FU2      SET    FU1+1                                                   PROSEQ    1991      I
          -CCG5
PROSEQ             IFC    NE,/"S"/FU1/,1                                          PROSEQ    1992      I
          -CCG5
PROSEQ    FU2      SET    /FUNIT/"S"                                              PROSEQ    1993      I
          -CCG5
PROSEQ    #MD      ENDIF                                            '6600         PROSEQ    1994      I
          -CCG5
PROSEQ    *                       INSTRUCTION SIZE                               PROSEQ    1995      I
          -CCG5
PROSEQ    S        GETARG 54,2,(SS)                                               PROSEQ    1996      I
          -CCG5
PROSEQ    PA       SET    "S" 1                                                   PROSEQ    1997      I
          -CCG5
PROSEQ    PA       SET    PA/15                                                   PROSEQ    1998      I
          -CCG5
PROSEQ    S        GETARG 63,6,(SS)                                               PROSEQ    1999      I
          -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            IFC    EQ,/"S"/ZEROP/,1                                    PROSEQ    2000    I
          -CCG5
PROSEQ     CO     SET    2                                                   PROSEQ    2001    I
          -CCG5
PROSEQ    *                        LOAD / STORE / JUMP BITS                 PROSEQ    2002    I
          -CCG5
PROSEQ            IF     DEF,Z."S",1                                         PROSEQ    2003    I
          -CCG5
PROSEQ     LSJ    SET    Z."S"                                               PROSEQ    2004    I
          -CCG5
PROSEQ    *                        ^D - NO RI / RI NOT A DEF                PROSEQ    2005    I
          -CCG5
PROSEQ            IFEQ   TYPE,3,2                                            PROSEQ    2006    I
          -CCG5
PROSEQ     BD     SET    1                                                   PROSEQ    2007    I
          -CCG5
PROSEQ            ELSE   2                                                   PROSEQ    2008    I
          -CCG5
PROSEQ            IF     DEF,Y."S",1                                         PROSEQ    2009    I
          -CCG5
PROSEQ     BD     SET    1                                                   PROSEQ    2010    I
          -CCG5
PROSEQ    *                        BOUNDARY MARKER                          PROSEQ    2011    I
          -CCG5
PROSEQ     S      GETARG 70,1,(SS)                                           PROSEQ    2012    I
          -CCG5
PROSEQ            IFC    NE,/"S"//,1                                         PROSEQ    2013    I
          -CCG5
PROSEQ     BD     SET    BD+2                                                PROSEQ    2014    I
          -CCG5
PROSEQ            IF     DEF,S.NAM,1                                         PROSEQ    2015    I
          -CCG5
PROSEQ     XSP    SET    S.NAM                                               PROSEQ    2016    I
          -CCG5
PROSEQ    *                                                                  PROSEQ    2017    I
          -CCG5
PROSEQ      VFD 2/TYPE,5/FT,2/1,4/FU1,4/FU2,2/BD,8/LSJ,3/XSP,2/JFT,3/CO,2/PA,23/  PROSEQ    2018    I
          -CCG5
PROSEQ     OPR    ENDM                                                        PROSEQ    2019    I
          -CCG5
PROSEQ     RDT    SPACE  3                                                   PROSEQ    2020    I
          -CCG5
PROSEQ    **       F.RDT - RLIST INSTRUCTION DESCRIPTOR TABLE                PROSEQ    2021    I
          -CCG5
PROSEQ                                                                       PROSEQ    2022    I
          -CCG5
PROSEQ            ENTRY  F.RDT                                               PROSEQ    2023    I
          -CCG5
PROSEQ     F.RDT  BSS    0                                                   PROSEQ    2024    I
          -CCG5
PROSEQ            LOC    0                                                   PROSEQ    2025    I
          -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ   *CALL     OPRDEFS                                                    PROSEQ      2026          I
         -CCG5
PROSEQ             LOC    *O                                                  PROSEQ      2027          I
         -CCG5
PROSEQ             LIST   R                                                   PROSEQ      2028          I
         -CCG5
PROSEQ    SST      TITLE  SST - SHELL SORT TABLE                              PROSEQ      2029          I
         -CCG1     -CCG5
PROSEQ   ***       SST - SHELL SORT TABLE.                                    PROSEQ      2030          I
         -CCG1     -CCG5
PROSEQ    SST=     TITLE  SST= - SHELL SORT TABLE                             CCG1         198          I
         -CCG5
PROSEQ   ***       SST= - SHELL SORT OF A SINGLE WORD PER ENTRY TABLE.        CCG1         199          I
         -CCG5
PROSEQ   *                                                                    PROSEQ      2031          I
         -CCG5
PROSEQ   *                SST SORTS A TABLE USING A SHELL SORTING TECHNIQUE.  PROSEQ      2032          I
         -CCG1     -CCG5
PROSEQ   *         SST= SORTS A TABLE USING A SHELL SORT TECHNIQUE.           CCG1         200          I
         -CCG5
PROSEQ   *         THE TABLE IS SORTED IN PLACE INTO ASCENDING ORDER ON THE KEY  PROSEQ      2033          I
         -CCG5
PROSEQ   *         KEY(X) = XOR(SHIFT(KEYMASK&X,KEYSHIFT),UPDOWN)             PROSEQ      2034          I
         -CCG5
PROSEQ   *                                                                    PROSEQ      2035          I
         -CCG5
PROSEQ   *                TIME(SST) @ TIME(RADIX) FOR N @ 250/575, B6 "0/=0  PROSEQ      2036          I
         -CCG5
PROSEQ   *                                                                    PROSEQ      2037          I
         -CCG5
PROSEQ   *         ENTRY  (B7) = ADDRESS OF TABLE TO BE SORTED                PROSEQ      2038          I
         -CCG5
PROSEQ   *                (B6) = SHIFT COUNT FOR EXTRACTED KEY PRIOR TO COMPARE  PROSEQ      2039          I
         -CCG5
PROSEQ   *                (X0) = MASK TO EXTRACT KEY FROM ENTRY              PROSEQ      2040          I
         -CCG5
PROSEQ   *                (X1) = NUMBER OF ELEMENTS IN ARRAY.                PROSEQ      2041          I
         -CCG5
PROSEQ   *                (X5) = ASCENDING/DESENDING SORT MASK ( +0 / -0 )   PROSEQ      2042          I
         -CCG5
PROSEQ   *                                                                    PROSEQ      2043          I
         -CCG5
PROSEQ   *         USES   X - ALL                                             PROSEQ      2044          I
         -CCG5
PROSEQ   *                B - 2, 3, 4, 5, 6                                  PROSEQ      2045          I
         -CCG5
PROSEQ   *                A - 1, 2, 6, 7.                                    PROSEQ      2046          I
         -CCG5
PROSEQ                                                                        PROSEQ      2047          I
         -CCG5
PROSEQ    SST0     SA2    B2          I = I + 1                              PROSEQ      2048          I
         -CCG5

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            SA7    B5-B4                                          PROSEQ    2049      I
          -CCG5
PROSEQ            EQ     B2,B3,SST4  IF I > N                           PROSEQ    2050      I
          -CCG5
PROSEQ                                                                  PROSEQ    2051      I
          -CCG5
PROSEQ    SST1    BX7    X0*X2                                          PROSEQ    2052      I
          -CCG5
PROSEQ            SA1    B2+B4       J = I - K                          PROSEQ    2053      I
          -CCG5
PROSEQ            LX6    B6,X7                                          PROSEQ    2054      I
          -CCG5
PROSEQ            SB5    B2+B4                                          PROSEQ    2055      I
          -CCG5
PROSEQ            BX3    X6-X5       SI = KEY(T(I))                     PROSEQ    2056      I
          -CCG5
PROSEQ            SB2    B2+1                                           PROSEQ    2057      I
          -CCG5
PROSEQ            LX7    X2                                             PROSEQ    2058      I
          -CCG5
PROSEQ                                                                  PROSEQ    2059      I
          -CCG5
PROSEQ    SST2    BX2    X0*X1                                          PROSEQ    2060      I
          -CCG5
PROSEQ            LX6    B6,X2                                          PROSEQ    2061      I
          -CCG5
PROSEQ            BX2    X6-X5       SJ = KEY(T(J))                     PROSEQ    2062      I
          -CCG5
PROSEQ            IX6    X3-X2                                          PROSEQ    2063      I
          -CCG5
PROSEQ            PL     X6,SST0     IF ELEMENTS IN ORDER               PROSEQ    2064      I
          -CCG5
PROSEQ            BX6    X1                                             PROSEQ    2065      I
          -CCG5
PROSEQ            SA1    B5+B4       J = J - K                          PROSEQ    2066      I
          -CCG5
PROSEQ            SA6    B5-B4       T(J+K) = T(J)                      PROSEQ    2067      I
          -CCG5
PROSEQ            SB5    B5+B4                                          PROSEQ    2068      I
          -CCG5
PROSEQ            GE     B5,B7,SST2  IF J \ 0                           PROSEQ    2069      I
          -CCG5
PROSEQ            SA2    B2          I = I + 1                          PROSEQ    2070      I
          -CCG5
PROSEQ            SA7    B5-B4       T(J+K) = S                         PROSEQ    2071      I
          -CCG5
PROSEQ            LT     B2,B3,SST1  IF I @ LENGTH                      PROSEQ    2072      I
          -CCG5
PROSEQ                                                                  PROSEQ    2073      I
          -CCG5
PROSEQ    SST4    AX4    1           K = K/2                            PROSEQ    2074      I
          -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            SB4    X4                                          PROSEQ     2075        I
          -CCG5
PROSEQ            SA2    B7-B4       I = K                           PROSEQ     2076        I
          -CCG5
PROSEQ            SB2    B7-B4                                       PROSEQ     2077        I
          -CCG5
PROSEQ            NZ     X4,SST1     IF K " 0                        PROSEQ     2078        I
          -CCG5
PROSEQ    SST     ENTRY.                                             PROSEQ     2079        I
          -CCG1   -CCG5
PROSEQ    SST=    ENTRY.                                             CCG1        201        I
          -CCG5
PROSEQ            MX4    12                                          PROSEQ     2080        I
          -CCG5
PROSEQ            NX6    B2,X1       K = 2**FIX(LOG2(LENGTH))+1      PROSEQ     2081        I
          -CCG5
PROSEQ            SB1    1                                           PROSEQ     2082        I
          -CCG5
PROSEQ            SB3    B7+X1       (B3) = LWA + 1                  PROSEQ     2083        I
          -CCG5
PROSEQ            AX4    B2,X4                                       PROSEQ     2084        I
          -CCG5
PROSEQ            NZ     B6,SST4     IF SHFTC " 0                    PROSEQ     2085        I
          -CCG5
PROSEQ                                                               PROSEQ     2086        I
          -CCG5
PROSEQ            QUAL   SHFTC=0                                     PROSEQ     2087        I
          -CCG5
PROSEQ                                                               PROSEQ     2088        I
          -CCG5
PROSEQ    SST0    AX4    1                                           PROSEQ     2089        I
          -CCG5
PROSEQ            SB4    X4          K = K/2                         PROSEQ     2090        I
          -CCG5
PROSEQ            SA2    B7-B4                                       PROSEQ     2091        I
          -CCG5
PROSEQ            SB2    B7-B4       I = K                           PROSEQ     2092        I
          -CCG5
PROSEQ            NZ     X4,SST2     IF K " 0                        PROSEQ     2093        I
          -CCG5
PROSEQ            EQ     SST                                         PROSEQ     2094        I
          -CCG1   -CCG5
PROSEQ            EQ     SST=                                        CCG1        202        I
          -CCG5
PROSEQ                                                               PROSEQ     2095        I
          -CCG5
PROSEQ    SST1    SA2    A2+B1       I = I + 1                       PROSEQ     2096        I
          -CCG5
PROSEQ            SA7    B5-B4       T(J+K) = S                      PROSEQ     2097        I
          -CCG5
PROSEQ            EQ     B2,B3,SST0  IF I > N                        PROSEQ     2098        I
          -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ                                                                                  PROSEQ      2099          I
          -CCG5
PROSEQ     SST2     BX6     X0*X2                                                       PROSEQ      2100          I
          -CCG5
PROSEQ              SA1     B2+B4       J = I - K                                       PROSEQ      2101          I
          -CCG5
PROSEQ              LX7     X2          S = T(I)                                        PROSEQ      2102          I
          -CCG5
PROSEQ              SB5     B2+B4                                                       PROSEQ      2103          I
          -CCG5
PROSEQ              BX3     X6-X5       SI = KEY(T(I))                                  PROSEQ      2104          I
          -CCG5
PROSEQ              SB2     B2+1                                                        PROSEQ      2105          I
          -CCG5
PROSEQ              BX2     X0*X1                                                       PROSEQ      2106          I
          -CCG5
PROSEQ                                                                                  PROSEQ      2107          I
          -CCG5
PROSEQ     SST3     BX2     X2-X5                                                       PROSEQ      2108          I
          -CCG5
PROSEQ              IX6     X3-X2                                                       PROSEQ      2109          I
          -CCG5
PROSEQ              PL      X6,SST1     IF ELEMENTS IN ORDER                            PROSEQ      2110          I
          -CCG5
PROSEQ              LX6     X1                                                          PROSEQ      2111          I
          -CCG5
PROSEQ              SA1     B5+B4                                                       PROSEQ      2112          I
          -CCG5
PROSEQ              SA6     B5-B4       T(J+K) = T(J)                                   PROSEQ      2113          I
          -CCG5
PROSEQ              SB5     B5+B4       J = J - K                                       PROSEQ      2114          I
          -CCG5
PROSEQ              BX2     X0*X1                                                       PROSEQ      2115          I
          -CCG5
PROSEQ              GE      B5,B7,SST3                                                  PROSEQ      2116          I
          -CCG5
PROSEQ              SA2     A2+B1       I = I + 1                                       PROSEQ      2117          I
          -CCG5
PROSEQ              SA7     B5-B4                                                       PROSEQ      2118          I
          -CCG5
PROSEQ              LT      B2,B3,SST2  IF I @ N                                        PROSEQ      2119          I
          -CCG5
PROSEQ              EQ      SST0                                                        PROSEQ      2120          I
          -CCG5
PROSEQ                                                                                  PROSEQ      2121          I
          -CCG5
PROSEQ              QUAL                                                                PROSEQ      2122          I
          -CCG5
PROSEQ     SHL      TITLE   SHL - SHELL SORT                                            PROSEQ      2123          I
          -CCG1     -CCG5
PROSEQ    ***       SHL - SHELL SORT FOR 1 WORD/ENTRY TABLES                            PROSEQ      2124          I
          -CCG1     -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ    SHL=     TITLE  SHL= - SHELL SORT                                    CCG1        203          I
          -CCG5
PROSEQ    **       SHL= - SHELL SORT FOR 1 WORD PER ENTRY TABLES.              CCG1        204          I
          -CCG5
PROSEQ    *        INPLACE SORT OF TABLE INTO ASCENDING ORDER                  PROSEQ     2125          I
          -CCG5
PROSEQ    *                                                                    PROSEQ     2126          I
          -CCG5
PROSEQ    *               TIME(SHL) @ TIME(RADIX) FOR N @ 1650                 PROSEQ     2127          I
          -CCG5
PROSEQ    *                                                                    PROSEQ     2128          I
          -CCG5
PROSEQ    *        ENTRY  (X1) = N = TABLE LENGTH                              PROSEQ     2129          I
          -CCG5
PROSEQ    *               (B7) = TBL = = FWA OF ARRAY TO BE SORTED             PROSEQ     2130          I
          -CCG5
PROSEQ    *                                                                    PROSEQ     2131          I
          -CCG5
PROSEQ    *        EXIT   (B3) = LWA+1 OF TABLE                                PROSEQ     2132          I
          -CCG5
PROSEQ    *               (B7) = FWA                                           PROSEQ     2133          I
          -CCG5
PROSEQ    *                                                                    PROSEQ     2134          I
          -CCG5
PROSEQ    *        USES   X - 1, 2, 3, 4, 6, 7                                 PROSEQ     2135          I
          -CCG5
PROSEQ    *               B - 2, 3, 4, 5                                       PROSEQ     2136          I
          -CCG5
PROSEQ    *               A - 1, 2, 6, 7                                       PROSEQ     2137          I
          -CCG5
PROSEQ                                                                         PROSEQ     2138          I
          -CCG5
PROSEQ     SHL1     SA2    A2+B1       I = I + 1                               PROSEQ     2139          I
          -CCG5
PROSEQ              SA7    B5-B4       T(J+K) = S                              PROSEQ     2140          I
          -CCG5
PROSEQ              EQ     B2,B3,SHL4  IF I > N                                PROSEQ     2141          I
          -CCG5
PROSEQ                                                                         PROSEQ     2142          I
          -CCG5
PROSEQ     SHL2     SA1    B2+B4       J = I - K                               PROSEQ     2143          I
          -CCG5
PROSEQ              LX7    X2          S = T(I)                                PROSEQ     2144          I
          -CCG5
PROSEQ              SB5    B2+B4                                               PROSEQ     2145          I
          -CCG5
PROSEQ              SB2    B2+B1                                               PROSEQ     2146          I
          -CCG5
PROSEQ                                                                         PROSEQ     2147          I
          -CCG5
PROSEQ     SHL3     IX3    X2-X1                                               PROSEQ     2148          I
          -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            BX6    X1                                                PROSEQ     2149        I
          -CCG5
PROSEQ            PL     X3,SHL1     IF ELEMENTS IN ORDER                  PROSEQ     2150        I
          -CCG5
PROSEQ            SA1    A1+B4                                             PROSEQ     2151        I
          -CCG5
PROSEQ            NO                                                       PROSEQ     2152        I
          -CCG5
PROSEQ            SA6    B5-B4       T(J+K) = T(J)                         PROSEQ     2153        I
          -CCG5
PROSEQ            SB5    B5+B4       J = J - K                             PROSEQ     2154        I
          -CCG5
PROSEQ            GE     B5,B7,SHL3  IF J \ 0                              PROSEQ     2155        I
          -CCG5
PROSEQ            EQ     SHL1                                              PROSEQ     2156        I
          -CCG5
PROSEQ                                                                    PROSEQ     2157        I
          -CCG5
PROSEQ    SHL4    AX4    1                                                 PROSEQ     2158        I
          -CCG5
PROSEQ            SB4    X4          K = K/2                               PROSEQ     2159        I
          -CCG5
PROSEQ            SA2    B7-B4       I = K                                 PROSEQ     2160        I
          -CCG5
PROSEQ            SB2    B7-B4                                             PROSEQ     2161        I
          -CCG5
PROSEQ            NZ     X4,SHL2     IF K " 0                              PROSEQ     2162        I
          -CCG5
PROSEQ                                                                    PROSEQ     2163        I
          -CCG5
PROSEQ    SHL     ENTRY.                                                  PROSEQ     2164        I
          -CCG1   -CCG5
PROSEQ    SHL=    ENTRY.                                                  CCG1        205        I
          -CCG5
PROSEQ            MX4    12                                                PROSEQ     2165        I
          -CCG5
PROSEQ            SB1    1                                                 PROSEQ     2166        I
          -CCG5
PROSEQ            NX6    B2,X1       K = 2**(FIX(LOG2(N))+1)               PROSEQ     2167        I
          -CCG5
PROSEQ            SB3    B7+X1       (B3) = LWA+1                          PROSEQ     2168        I
          -CCG5
PROSEQ            AX4    B2,X4                                             PROSEQ     2169        I
          -CCG5
PROSEQ            EQ     SHL4                                              PROSEQ     2170        I
          -CCG5
PROSEQ    TABLES  TITLE  PASS 2 TABLES                                     PROSEQ     2171        I
          -CCG5
PROSEQ            USE    //                                                PROSEQ     2172        I
          -CCG5
PROSEQ    F.MEM   BSS    1           FWA OF WORKING STORAGE                PROSEQ     2173        I
          -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            USE    0                                                    PROSEQ    2174    I
          -CCG5
PROSEQ                                                                        PROSEQ    2175    I
          -CCG5
PROSEQ    **      TABLE - GENERATE MANAGED TABLE.                             PROSEQ    2176    I
          -CCG5
PROSEQ    *                                                                   PROSEQ    2177    I
          -CCG5
PROSEQ    *       TABLE  TNAME,EQUIV                                          PROSEQ    2178    I
          -CCG5
PROSEQ    *       ENTRY  *TNAME* = TABLE NAME.                                PROSEQ    2179    I
          -CCG5
PROSEQ    *              *EQUIV* = EQUIVALENT TABLE NAME.  THIS ALLOWS CERTAIN PROSEQ  2180    I
          -CCG5
PROSEQ    *                         TABLES TO BE USED BY DIFFERENT PROCESSORS.  PROSEQ    2181    I
          -CCG5
PROSEQ    *       EXIT   *O.TNAM* IS NAME OF WORD CONTAINING TABLE FWA.       PROSEQ    2182    I
          -CCG5
PROSEQ    *              *L.TNAM* IS NAME OF WORD CONTAINING THE TABLE LENGTH. PROSEQ  2183    I
          -CCG5
PROSEQ    *                         *Z.TNAM* IS THE TABLE NUMBER              PROSEQ    2184    I
          -CCG5
PROSEQ                                                                        PROSEQ    2185    I
          -CCG5
PROSEQ            PURGMAC TABLE                                               PROSEQ    2186    I
          -CCG5
PROSEQ            MACRO  TABLE,TNAM,EQUIV                                     PROSEQ    2187    I
          -CCG5
PROSEQ            ENTRY  O.TNAM,L.TNAM,Z.TNAM                                 PROSEQ    2188    I
          -CCG5
PROSEQ    T       IFC    EQ,/EQUIV//                                          PROSEQ    2189    I
          -CCG5
PROSEQ     Z.TNAM EQU    *-FTAB                                               PROSEQ    2190    I
          -CCG5
PROSEQ     O.TNAM CON    F.MEM                                                PROSEQ    2191    I
          -CCG5
PROSEQ     TABLES RMT                                                          PROSEQ    2192    I
          -CCG5
PROSEQ            ORG    LTAB+Z.TNAM                                          PROSEQ    2193    I
          -CCG5
PROSEQ     L.TNAM CON    0                                                    PROSEQ    2194    I
          -CCG5
PROSEQ     TABLES RMT                                                          PROSEQ    2195    I
          -CCG5
PROSEQ     TABNAM RMT                                                          PROSEQ    2196    I
          -CCG5
PROSEQ            ORG    NAMTAB+Z.TNAM                                        PROSEQ    2197    I
          -CCG5
PROSEQ            DATA 10H TNAM                                               PROSEQ    2198    I
          -CCG5
PROSEQ     TABNAM RMT                                                          PROSEQ    2199    I
          -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ   T         ELSE                                                    PROSEQ      2200          I
         -CCG5
PROSEQ    Z.TNAM   EQU    Z.EQUIV                                          PROSEQ      2201          I
         -CCG5
PROSEQ    O.TNAM   EQU    O.EQUIV                                          PROSEQ      2202          I
         -CCG5
PROSEQ    TABLES   RMT                                                     PROSEQ      2203          I
         -CCG5
PROSEQ    L.TNAM   EQU    L.EQUIV                                          PROSEQ      2204          I
         -CCG5
PROSEQ    TABLES   RMT                                                     PROSEQ      2205          I
         -CCG5
PROSEQ    TABNAM   RMT                                                     PROSEQ      2206          I
         -CCG5
PROSEQ             ORG    NAMTAB+Z.TNAM                                    PROSEQ      2207          I
         -CCG5
PROSEQ             DATA   10H TNAM EQUIV                                   PROSEQ      2208          I
         -CCG5
PROSEQ    TABNAM   RMT                                                     PROSEQ      2209          I
         -CCG5
PROSEQ   T         ENDIF                                                   PROSEQ      2210          I
         -CCG5
PROSEQ             ENDM                                                    PROSEQ      2211          I
         -CCG5
PROSEQ    TABLES   SPACE  3                                                PROSEQ      2212          I
         -CCG5
PROSEQ   **        PASS 2 MANAGED TABLE DEFINITIONS                        PROSEQ      2213          I
         -CCG5
PROSEQ                                                                     PROSEQ      2214          I
         -CCG5
PROSEQ    FTAB     BSS    0           FWA VECTOR FOR TABLE MANAGER         PROSEQ      2215          I
         -CCG5
PROSEQ                                                                     PROSEQ      2216          I
         -CCG5
PROSEQ    BLK      TABLE              BLOCKS ( OPT=2 ONLY )                PROSEQ      2217          I
         -CCG5
PROSEQ                                                                     PROSEQ      2218          I
         -CCG5
PROSEQ    TXT      TABLE              RLIST INSTRUCTIONS                   PROSEQ      2219          I
         -CCG5
PROSEQ    GST      TABLE  TXT         GRAPH STRUCTURE TABLE                PROSEQ      2220          I
         -CCG5
PROSEQ                                                                     PROSEQ      2221          I
         -CCG5
PROSEQ    RND      TABLE              R-NUMBER DEFINITION TABLE ( *SQZ* )  PROSEQ      2222          I
         -CCG5
PROSEQ    TREE     TABLE  RND         DEPENDENCY TREE / SUCCESSOR INDEX TABLE  PROSEQ  2223          I
         -CCG5
PROSEQ                                                                     PROSEQ      2224          I
         -CCG5
PROSEQ    PIT      TABLE              POSTED INSTRUCTIONS                  PROSEQ      2225          I
         -CCG5

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ                                                                                  PROSEQ      2226          I
          -CCG5
PROSEQ     OTI      TABLE              OT. INFO 12/P(OT. ORD),30/0,18/RI IN OT.         PROSEQ      2227          I
          -CCG5
PROSEQ     MLT      TABLE  OTI         MOD LIST INDEX TABLE                             PROSEQ      2228          I
          -CCG5
PROSEQ                                                                                  PROSEQ      2229          I
          -CCG5
PROSEQ     MOD      TABLE              MODIFICATIONS ( INSTRUCTIONS )                   PROSEQ      2230          I
          -CCG5
PROSEQ                                                                                  PROSEQ      2231          I
          -CCG5
PROSEQ     CFT      TABLE              CONTROL FLOW TABLE ( EDGES )                     PROSEQ      2232          I
          -CCG5
PROSEQ                                                                                  PROSEQ      2233          I
          -CCG5
PROSEQ     BST      TABLE              BLOCK STATUS INFORMATION                         PROSEQ      2234          I
          -CCG5
PROSEQ    * BIT     =      BST         ( PHASE 1 OF OPT=2 )                             PROSEQ      2235          I
          -CCG5
PROSEQ                                                                                  PROSEQ      2236          I
          -CCG5
PROSEQ     PSI      TABLE              POST STORE INFORMATION ( *GPO/GRA* )             PROSEQ      2237          I
          -CCG5
PROSEQ                                                                                  PROSEQ      2238          I
          -CCG5
PROSEQ     RXI      TABLE              REGION EXIT INFORMATION ( *GPO/GRA* )            PROSEQ      2239          I
          -CCG5
PROSEQ                                                                                  PROSEQ      2240          I
          -CCG5
PROSEQ     RCT      TABLE              REGISTER CANIDATE TABLE ( *GRA* )                PROSEQ      2241          I
          -CCG5
PROSEQ                                                                                  PROSEQ      2242          I
          -CCG5
PROSEQ     TET      TABLE              TEMP EQUIVALENCE TABLE ( CODE MOTION )           PROSEQ      2243          I
          -CCG5
PROSEQ                                                                                  PROSEQ      2244          I
          -CCG5
PROSEQ     IOL      TABLE              I/O AP LIST PARAMETER ORDINALS ( OPT=2 )         PROSEQ      2245          I
          -CCG5
PROSEQ     IIT      TABLE  IOL         INCREMENT INFORMATION ( OPT=2 )                  PROSEQ      2246          I
          -CCG5
PROSEQ                                                                                  PROSEQ      2247          I
          -CCG1     -CCG5
PROSEQ     VDT      TABLE              VARDIM INFO                                      PROSEQ      2248          I
          -CCG1     -CCG5
PROSEQ                                                                                  PROSEQ      2249          I
          -CCG1     -CCG5
PROSEQ     APL      TABLE              AP LISTS                                         PROSEQ      2250          I
          -CCG1     -CCG5
PROSEQ                                                                                  PROSEQ      2251          I
          -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ    UDI      TABLE              USE/DEF INDEX TABLE                          PROSEQ      2252          I
          -CCG5
PROSEQ                                                                            PROSEQ      2253          I
          -CCG5
PROSEQ    ALS      TABLE              24/,36/CA,IH OF ACTIVE LOOP *CV"S*           PROSEQ      2254          I
          -CCG1    -CCG5
PROSEQ             IFC    NE,//"HC.HDT"/,2                                         CCG1         206          I
          -CCG5
PROSEQ             ECHO   1,Z=("HC.HDT")                                           CCG1         207          I
          -CCG5
PROSEQ    Z        TABLE                                                           CCG1         208          I
          -CCG5
PROSEQ                                                                            CCG1         209          I
          -CCG5
PROSEQ    CVT      TABLE              CONSTANT VALUE TABLE                         CCG1         210          I
          -CCG5
PROSEQ                                                                            CCG1         211          I
          -CCG5
PROSEQ    CUT      TABLE              CONSTANT USED TABLE                          CCG1         212          I
          -CCG5
PROSEQ                                                                            CCG1         213          I
          -CCG5
PROSEQ    N.DT     EQU    *-FTAB      N. DYNAMIC TABLES                            CCG1         214          I
          -CCG5
PROSEQ                                                                            CCG1         215          I
          -CCG5
PROSEQ             ECHO   1,Z=("HC.FRTP")                                          CCG1         216          I
          -CCG5
PROSEQ    ;AT      TABLE               ;A ADDRESS DEFINITION TABLE                 CCG1         217          I
          -CCG5
PROSEQ    SST      TABLE              SST - SET SYMBOL VALUE TABLE                 CCG1         218          I
          -CCG5
PROSEQ                                                                            CCG1         219          I
          -CCG5
PROSEQ    CBT      TABLE              COMMON BLOCK NAME AND LENGTH TABLE           CCG1         220          I
          -CCG5
PROSEQ    *                           60/0L_NAME,2/LVL,34/,24/LENGTH               CCG1         221          I
          -CCG5
PROSEQ                                                                            CCG1         222          I
          -CCG5
PROSEQ    LBT      TABLE              LOCAL BLOCK NAME AND LENGTH TABLE            CCG1         223          I
          -CCG5
PROSEQ    *                           42/7L_NAME,18/LENGTH                         CCG1         224          I
          -CCG5
PROSEQ                                                                            CCG1         225          I
          -CCG5
PROSEQ             IFNE   HC.QBT,0,1                                               CCG1         226          I
          -CCG5
PROSEQ    QBT      TABLE              QUAL BLOCK NAME TABLE                        CCG1         227          I
          -CCG5
PROSEQ                                                                            CCG1         228          I
          -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            IFC    NE,/"HC.MCT"//,2                                  CCG1        229        I
          -CCG5
PROSEQ            ECHO   1,Z=("HC.MCT")                                    CCG1        230        I
          -CCG5
PROSEQ    Z       TABLE             Z TABLE                                CCG1        231        I
          -CCG5
PROSEQ                                                                     CCG1        232        I
          -CCG5
PROSEQ    SYM     TABLE             SYMBOL TABLE                           CCG1        233        I
          -CCG5
PROSEQ                                                                     PROSEQ     2255        I
          -CCG5
PROSEQ    TEND    TABLE             DUMMY TABLE FOR TABLE MANAGER          PROSEQ     2256        I
          -CCG5
PROSEQ            BSS    0                                                 PROSEQ     2257        I
          -CCG5
PROSEQ    NTAB    EQU    *-FTAB     TOTAL NUMBER OF TABLES                PROSEQ     2258        I
          -CCG5
PROSEQ            SPACE  3                                                 PROSEQ     2259        I
          -CCG5
PROSEQ    **      REMAINDER OF TABLE MANAGER STORAGE                       PROSEQ     2260        I
          -CCG5
PROSEQ                                                                     PROSEQ     2261        I
          -CCG5
PROSEQ    LTAB    BSS    0           TABLE LENGTHS                         PROSEQ     2262        I
          -CCG5
PROSEQ    TABLES  HERE                                                     PROSEQ     2263        I
          -CCG5
PROSEQ                                                                     PROSEQ     2264        I
          -CCG5
PROSEQ    MU      ENTRY. 0           MEMORY USED                           PROSEQ     2265        I
          -CCG1   -CCG5
PROSEQ    LM      ENTRY. F.MEM       LOW MEMORY ADDRESS ( FWA WORKING STORAGE )  PROSEQ  2266     I
          -CCG5
PROSEQ    TN      CON    NTAB        NUMBER OF MANAGED TABLES              PROSEQ     2267        I
          -CCG1   -CCG5
PROSEQ    TN      CON    N.DT        NUMBER OF ACTIVE TABLES               CCG1        234        I
          -CCG5
PROSEQ            BSS    4           SCRATCH TO SAVE - X0 , X5 , X1 , A0   PROSEQ     2268        I
          -CCG5
PROSEQ    PL      CON    0           PREVIOUS LENGTH OF REQUESTING TABLE   PROSEQ     2269        I
          -CCG5
PROSEQ    TO      ENTRY. TOV         ADDRESS OF TABLE OVERFLOW PROCESSOR   PROSEQ     2270        I
          -CCG5
PROSEQ    INT.P2M ENTRY.             */ FIELD LENGTH AT THE BEG.OF PASS2   RSM2763X     170        I
          -CCG5
PROSEQ    MX.BLN  ENTRY.                                                   RSM2763X     171        I
          -CCG5
PROSEQ    TOBA    ENTRY.                                                   RSM2763X     172        I
          -CCG5
PROSEQ    TOBB    BSS    1                                                 RSM2763X     173        I
          -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ    TOV      SPACE  3                                                    PROSEQ      2271          I
          -CCG5
PROSEQ    **       TOV - TABLE OVERFLOW PROCESSOR                              PROSEQ      2272          I
          -CCG5
PROSEQ                                                                         PROSEQ      2273          I
          -CCG5
PROSEQ             ENTRY  TOV                                                  PROSEQ      2274          I
          -CCG5
PROSEQ    TOV      BSS    0                                                    PROSEQ      2275          I
          -CCG5
PROSEQ             SB5    -B5                                                  RSM2763X     174          I
          -CCG5
PROSEQ             SX1    B5+2000B    WR = NO.OF WORDS REQUESTED               RSM2763X     175          I
          -CCG5
PROSEQ             PRINT  TOV,(* TOV CALLS MOREFL,WR=*Z7),(X1)                  RSM2763X     176          I
          -CCG5
PROSEQ             CALL   MOREFL                                               RSM2763X     177          I
          -CCG5
PROSEQ             PL     X7,/TBLM/ATS9                                        RSM2763X     178          I
          -CCG5
PROSEQ    .T       IFNE   TEST,0                              * TEST MODE *    PROSEQ      2276          I
          -CCG5
PROSEQ             SA5    ATS=                                                 PROSEQ      2277          I
          -CCG5
PROSEQ             LX5    30                                                   PROSEQ      2278          I
          -CCG5
PROSEQ             DEBUG                                                       PROSEQ      2279          I
          -FEAT155F -CCG5
PROSEQ             SNAP   FTAB,TO,,1,10,1,TOV                                  PROSEQ      2280          I
          -FEAT155F -CCG5
PROSEQ    TOV      SNAP   FTAB,TO                                              FEAT155F     257          I
          -CCG5
PROSEQ    .T       ENDIF                                      * TEST MODE *    PROSEQ      2281          I
          -CCG5
PROSEQ                                                                         PROSEQ      2282          I
          -CCG5
PROSEQ             SA1    =7LTBL-OVF                                           PROSEQ      2283          I
          -RSM2763X -CCG5
PROSEQ    TOV1     SA1    =7LTBL-OVF                                           RSM2763X     179          I
          -CCG5
PROSEQ             CALL   PUNT        ISSUE AN ERROR MESSAGE AND QUIT          PROSEQ      2284          I
          -CCG5
PROSEQ    TOB      SPACE  3,14                                                 PROSEQ      2285          I
          -CCG5
PROSEQ    **       TOB - OVERFLOW PROCESSING FOR PASS2,OPT2                    RSM2763X     180          I
          -CCG5
PROSEQ    *                                                                    RSM2763X     181          I
          -CCG5
PROSEQ    *        ENTRY (B5) = - NO.OF WORDS NEEDED                           RSM2763X     182          I
          -CCG5
PROSEQ    *                                                                    RSM2763X     183          I
          -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ   *         TOB ATTEMPTS TO OBTAIN NEEDED WORDS BY DECREASING BLK TABLE          RSM2763X    184        I
         -CCG5
PROSEQ   *         IF IT IS AT LEAST 1.5 TIMES THE MAX.BASIC BLOCK SIZE,ELSE            RSM2763X    185        I
         -CCG5
PROSEQ   *         TOB CALLS MOREFL TO GET ADDIT.SPACE BY EXTENDING FIELD               RSM2763X    186        I
         -CCG5
PROSEQ   *         LENGTH. IF NO FL AVAIL.,COMPILATION IS ABORTED                       RSM2763X    187        I
         -CCG5
PROSEQ                                                                                  RSM2763X    188        I
         -CCG5
PROSEQ             ENTRY  TOB                                                           RSM2763X    189        I
         -CCG5
PROSEQ    TOB      SA2    =XMX.BLN                                                      RSM2763X    190        I
         -CCG5
PROSEQ             SA3    =XM.BBL                                                       RSM2763X    191        I
         -CCG5
PROSEQ             AX2    1           X2 = MX.BLN/2                                     RSM2763X    192        I
         -CCG5
PROSEQ             BX1    X3                                                            RSM2763X    193        I
         -CCG5
PROSEQ             AX1    1                                                             RSM2763X    194        I
         -CCG5
PROSEQ             IX1    X1+X3       X1 = 1.5*M.BBL                                    RSM2763X    195        I
         -CCG5
PROSEQ             IX4    X2-X1                                                         RSM2763X    196        I
         -CCG5
PROSEQ             PL     X4,TOB1     IF MX.BLN/2 > 1.5*M.BBL                           RSM2763X    197        I
         -CCG5
PROSEQ             BX2    X1                                                            RSM2763X    198        I
         -CCG5
PROSEQ                                                                                  RSM2763X    199        I
         -CCG5
PROSEQ   *         MIN.BLK = MAX(MX.BLN/2,1.5*M.BBL)                                    RSM2763X    200        I
         -CCG5
PROSEQ                                                                                  RSM2763X    201        I
         -CCG5
PROSEQ    TOB1     SB5    -B5         WN = NO.OF WORDS NEEDED                           RSM2763X    202        I
         -CCG5
PROSEQ             PRINT  TOB,(*  NO.OF WORDS NEEDED,MIN.BLK = *2Z7),(B5,X1)             RSM2763X    203        I
         -CCG5
PROSEQ             SA3    L.BLK                                                         RSM2763X    204        I
         -CCG5
PROSEQ             SX1    B5+200B     WR = WN+200B */ WORDS TO BE REQUESTED             RSM2763X    205        I
         -CCG5
PROSEQ             IX4    X3-X2       WA = L.BLK - MIN.BLK */ WORDS AVAIL.IN BLK        RSM2763X    206        I
         -CCG5
PROSEQ             SB2    B1          SPECIAL PROC.BY DMB IF MOREFL NOT CALLED          RSM2763X    207        I
         -CCG5
PROSEQ             IX2    X4-X1                                                         RSM2763X    208        I
         -CCG5
PROSEQ             BX6    X1                                                            RSM2763X    209        I
         -CCG5

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            PL     X2,TOB2     IF WA > WR                                  RSM2763X    210        I
          -CCG5
PROSEQ            CALL   MOREFL      MOREFL(X6) = WG */ WORDS GRANTED IN NEW FL  RSM2763X    211        I
          -CCG5
PROSEQ            PL     X7,/TBLM/ATS9    IF WG > WR                             RSM2763X    212        I
          -CCG5
PROSEQ                                                                           RSM2763X    213        I
          -CCG5
PROSEQ            IX6    X1-X6       WR = WR-WG                                  RSM2763X    214        I
          -CCG5
PROSEQ            SB2    B0          STAND.PROC BY DMB IF MOREFL CALLED          RSM2763X    215        I
          -CCG5
PROSEQ                                                                           RSM2763X    216        I
          -CCG5
PROSEQ    *       GET REMAINING NEEDED SPACE BY REDUCING *BLK* SIZE              RSM2763X    217        I
          -CCG5
PROSEQ                                                                           RSM2763X    218        I
          -CCG5
PROSEQ     TOB2   SA2    =XMX.AVS                                                RSM2763X    219        I
          -CCG5
PROSEQ            IX1    X6-X2       WD = WR - MX.AVS */ WORDS TO BE DUMPED      RSM2763X    220        I
          -CCG5
PROSEQ            SB2    X1                                                      RSM2763X    221        I
          -CCG5
PROSEQ            LE     B2,B0,TOB3  IF WD .LE. ZERO                             RSM2763X    222        I
          -CCG5
PROSEQ            SA6    TOBA        SAVE WR                                     RSM2763X    223        I
          -CCG5
PROSEQ            PRINT  TOB,(*  TOB CALLS  DMB ; WDS,SP.FLG=*2Z7),(X1,B2)       RSM2763X    224        I
          -CCG5
PROSEQ            CALL   DMB         DUMP BLOCKS TO FREE WD WORDS                RSM2763X    225        I
          -CCG5
PROSEQ    *       DMB(X2) = WFL */ NO. OF WORDS BY MOREFL                        RSM2763X    226        I
          -CCG5
PROSEQ            NZ     X1,TOV1     IF X1 " 0 */ CANNOT DUMP ENOUGH BLOCKS      RSM2763X    227        I
          -CCG5
PROSEQ            SA1    TOBA        RELOAD WR                                   RSM2763X    228        I
          -CCG5
PROSEQ            IX6    X1-X2       BLK.DECR = WR - WFL                         RSM2763X    229        I
          -CCG5
PROSEQ            MI     X6,TOB4     IF BLK.DECR @ 0                             RSM2763X    230        I
          -CCG5
PROSEQ                                                                           RSM2763X    231        I
          -CCG5
PROSEQ    *       DECREASE BLOCK TABLE SIZE BY BLK.DECR                          RSM2763X    232        I
          -CCG5
PROSEQ                                                                           RSM2763X    233        I
          -CCG5
PROSEQ     TOB3   SA1    L.BLK                                                   RSM2763X    234        I
          -CCG5
PROSEQ            IX7    X1-X6                                                   RSM2763X    235        I
          -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            SA7    A1          L.BLK = L.BLK - WR                          RSM2763X     236          I
          -CCG5
PROSEQ            PRINT  TOB,(*  NEW BLK SIZE,DIFFERENCE =*2Z7),(X7,X6)          RSM2763X     237          I
          -CCG5
PROSEQ    TOB4    CALL   CGB         COLLECT FREE SPACE AT THE END OF BLK        RSM2763X     238          I
          -CCG5
PROSEQ            EQ     /TBLM/ATS9                                              RSM2763X     239          I
          -CCG5
PROSEQ                                                                           RSM2763X     240          I
          -CCG5
PROSEQ                                                                           RSM2763X     241          I
          -CCG5
PROSEQ    **      GET MORE MEMORY FROM THE SYSTEM                                RSM2763X     242          I
          -CCG5
PROSEQ    *                                                                      RSM2763X     243          I
          -CCG5
PROSEQ    *       ENTRY (X1) = WN ,  NO.OF WORDS NEEDED                          RSM2763X     244          I
          -CCG5
PROSEQ    *                                                                      RSM2763X     245          I
          -CCG5
PROSEQ    *       EXIT   (X1) = WN                                               RSM2763X     246          I
          -CCG5
PROSEQ    *              (X6) = WG , NO. OF WORDS GRANTED                        RSM2763X     247          I
          -CCG5
PROSEQ    *              (X7) = WG - WN                                          RSM2763X     248          I
          -CCG5
PROSEQ                                                                           RSM2763X     249          I
          -CCG5
PROSEQ    MOREFL  ENTRY.                                                         RSM2763X     250          I
          -CCG5
PROSEQ            PRINT  MOREFL,(*  WN =*Z7),(X1)                                RSM2763X     251          I
          -CCG5
PROSEQ            SA2    =XMAX.FL    MXFL = MAX. FL WE CAN POSSIBLY GET          RSM2763X     252          I
          -CCG5
PROSEQ            SA3    =XCP.AFLS   CFL = FL CURRENTLY USED                     RSM2763X     253          I
          -CCG5
PROSEQ            AX2    30                                                      RSM2763X     254          I
          -CCG5
PROSEQ            IX6    X2-X3       WA = NO. OF WORDS AVAILABLE IN MXFL         RSM2763X     255          I
          -CCG5
PROSEQ            MX7    44                                                      RSM2763X     256          I
          -CCG5
PROSEQ            ZR     X6,MOREFL   IF WA = 0 */ MAX FL USED ALREADY            RSM2763X     257          I
          -CCG5
PROSEQ            SB3    X6                                                      RSM2763X     258          I
          -CCG5
PROSEQ            MX7    -6                                                      RSM2763X     259          I
          -CCG5
PROSEQ            IX2    X1-X7                                                   RSM2763X     260          I
          -CCG5
PROSEQ            BX2    X7*X2       ROUND WN TO NEXT 100B                       RSM2763X     261          I
          -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            SB2    X2+1000B                                          RSM2763X    262        I
          -CCG5
PROSEQ            GE     B2,B3,MFL1                                        RSM2763X    263        I
          -CCG5
PROSEQ            SX6    B2          WR = MIN(WA,WN+1000B) */ NO OF WDS REQ. RSM2763X  264        I
          -CCG5
PROSEQ                                                                     RSM2763X    265        I
          -CCG5
PROSEQ     MFL1   IX7    X6+X3                                             RSM2763X    266        I
          -CCG5
PROSEQ            LX7    30                                                RSM2763X    267        I
          -CCG5
PROSEQ            SA7    =XGT1                                             RSM2763X    268        I
          -CCG5
PROSEQ            BX7    X1                                                RSM2763X    269        I
          -CCG5
PROSEQ            SA7    TOBB                                              RSM2763X    270        I
          -CCG5
PROSEQ            PRINT  MOREFL,(* WA,WR =*2Z7),(B3,X6)                    RSM2763X    271        I
          -CCG5
PROSEQ            MEMORY SCM,GT1,RGL                                       RSM2763X    272        I
          -CCG5
PROSEQ            SA2    GT1                                               RSM2763X    273        I
          -CCG5
PROSEQ            SA3    CP.AFLS     OFL = OLD FL                          RSM2763X    274        I
          -CCG5
PROSEQ            AX2    30          NFL = NEW FL                          RSM2763X    275        I
          -CCG5
PROSEQ            IX0    X2-X3       WG = NO. OF WORDS GRANTED             RSM2763X    276        I
          -CCG5
PROSEQ            BX6    X2                                                RSM2763X    277        I
          -CCG5
PROSEQ            SA6    A3          CP.AFLS = NEW FL                      RSM2763X    278        I
          -CCG5
PROSEQ            PRINT  MOREFL,(*  NEW FL,WG =*2Z7),(X6,X0)               RSM2763X    279        I
          -CCG5
PROSEQ            RJ     AST         */ ADJUST SPECIAL TABLES              RSM2763X    280        I
          -CCG5
PROSEQ            SA1    TOBB                                              RSM2763X    281        I
          -CCG5
PROSEQ            BX6    X0          (X6) = NO.OF WORDS GRANTED */ EXIT PAR. RSM2763X  282        I
          -CCG5
PROSEQ            IX7    X0-X1                                             RSM2763X    283        I
          -CCG5
PROSEQ            EQ     MOREFL                                            RSM2763X    284        I
          -CCG5
PROSEQ     CWS=   SPACE  3,14                                              RSM2763X    285        I
          -CCG5
PROSEQ    **      CWS= - CALCULATE WORKING STORAGE SIZE                    RSM2763X    286        I
          -CCG5
PROSEQ    *                                                               RSM2763X    287        I
          -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ   *         EXIT   (X6) = CURRENT WORKING STORAGE SIZE                    RSM2763X     288          I
         -CCG5
PROSEQ   *                (X7) = MAXIMUM WORKING STORAGE SIZE                    RSM2763X     289          I
         -CCG5
PROSEQ                                                                            RSM2763X     290          I
         -CCG5
PROSEQ    CWS=     ENTRY.                                                         RSM2763X     291          I
         -CCG5
PROSEQ             SA1    LM                                                      RSM2763X     292          I
         -CCG5
PROSEQ             SA2    O.TEND                                                  RSM2763X     293          I
         -CCG5
PROSEQ             SA3    =XCP.AFLS                                               RSM2763X     294          I
         -CCG5
PROSEQ             SA4    MAX.FL                                                  RSM2763X     295          I
         -CCG5
PROSEQ             AX4    30                                                      RSM2763X     296          I
         -CCG5
PROSEQ             IX6    X2-X1       CWS = O.TEND - LM                           RSM2763X     297          I
         -CCG5
PROSEQ             IX7    X4-X3                                                   RSM2763X     298          I
         -CCG5
PROSEQ             IX7    X7+X6       MWS = MAX.FL - CP.AFLS + CWS                RSM2763X     299          I
         -CCG5
PROSEQ             EQ     CWS=                                                    RSM2763X     300          I
         -CCG5
PROSEQ    AFL      SPACE  3,14                                                    RSM2763X     301          I
         -CCG5
PROSEQ   **        AFL - ADJUST *FL* AT END OF PASS 2 PROCESSING TO REDUCE IT    RSM2763X     302          I
         -CCG5
PROSEQ   **        TO ITS VALUE AT THE BEGINNING OF PASS 2.                      RSM2763X     303          I
         -CCG5
PROSEQ    AFL      ENTRY.                                                         RSM2763X     304          I
         -CCG5
PROSEQ             SA1    =XCP.AFLS                                               RSM2763X     305          I
         -CCG5
PROSEQ             SA2    INT.P2M                                                 RSM2763X     306          I
         -CCG5
PROSEQ             AX2    30                                                      RSM2763X     307          I
         -CCG5
PROSEQ             IX0    X2-X1                                                   RSM2763X     308          I
         -CCG5
PROSEQ             BX6    X1                                                      RSM2763X     309          I
         -CCG5
PROSEQ             SA6    =XPR.MXFL   PR.MXFL = CP.AFLS                           RSM2763X     310          I
         -CCG5
PROSEQ             PRINT  AFL,(* END OF PASS2,CP.AFLS,CHANGE=*2Z7),(X1,X0)       RSM2763X     311          I
         -CCG5
PROSEQ             ZR     X0,AFL      IF CURRENT FL = INT.P2 FL                   RSM2763X     312          I
         -CCG5
PROSEQ                                                                            RSM2763X     313          I
         -CCG5

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            RJ     AST         MOVE SPECIAL TABLES DOWN                          RSM2763X     314          I
          -CCG5
PROSEQ            MEMORY SCM,INT.P2M,RCL   REDUCE FL TO INITIAL VALUE                  RSM2763X     315          I
          -CCG5
PROSEQ            SA5    INT.P2M                                                       RSM2763X     316          I
          -CCG5
PROSEQ            SA1    CP.AFLS                                                       RSM2763X     317          I
          -CCG5
PROSEQ            AX5    30                                                            RSM2763X     318          I
          -CCG5
PROSEQ            SX6    X5                                                            RSM2763X     319          I
          -CCG5
PROSEQ            SA6    A1          CP.AFLS = INT.P2M                                 RSM2763X     320          I
          -CCG5
PROSEQ            SX0    X1-100000B                                                    RSM2763X     321          I
          -CCG5
PROSEQ            MI     X0,AFL            IF F < 100 000B                             RSM2763X     322          I
          -CCG5
PROSEQ                                                                                 RSM2763X     323          I
          -CCG5
PROSEQ    *       FL NEEDED \ 100 000B ISSUE A DAYFILE MESSAGE                         RSM2763X     324          I
          -CCG5
PROSEQ                                                                                 RSM2763X     325          I
          -CCG5
PROSEQ            CALL   COD                                                            RSM2763X     326          I
          -CCG1   -CCG5
PROSEQ            CALL   COD=                                                           CCG1         235          I
          -CCG5
PROSEQ            LX6    12                                                             RSM2763X     327          I
          -CCG5
PROSEQ            SX5    2R  -2RB                                                       RSM2763X     328          I
          -CCG5
PROSEQ            IX6    X6-X5                                                          RSM2763X     329          I
          -CCG5
PROSEQ            SA2    =XPROGNAM                                                      RSM2763X     330          I
          -CCG1   -CCG5
PROSEQ            SA2    =XCO.PRGN                                                      CCG1         236          I
          -CCG5
PROSEQ            SA3    =3R                                                            RSM2763X     331          I
          -CCG5
PROSEQ            BX7    X2+X3                                                          RSM2763X     332          I
          -CCG5
PROSEQ            SA6    AFLA+1                                                         RSM2763X     333          I
          -CCG5
PROSEQ            LX7    -18                                                            RSM2763X     334          I
          -CCG5
PROSEQ            SA7    A6-B1                                                          RSM2763X     335          I
          -CCG5
PROSEQ            MESSAGE AFLA,,R    MESSAGE( *   PROGNAM  NNNNNNB SCM USED*)           RSM2763X     336          I
          -CCG1   -CCG5
PROSEQ            MESSAGE AFLA,,R    MESSAGE( *   CO.PRGN  NNNNNNB SCM USED*)           CCG1         237          I
          -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            EQ     AFL                                                    RSM2763X     337          I
          -CCG5
PROSEQ                                                                          RSM2763X     338          I
          -CCG5
PROSEQ    AFLA    DATA   0,0,18LSCM USED                                        RSM2763X     339          I
          -CCG5
PROSEQ                                                                          RSM2763X     340          I
          -CCG5
PROSEQ                                                                          RSM2763X     341          I
          -CCG5
PROSEQ    **      AST - ADJUST SPECIAL TABLES IN HIGH CORE AND POINTERS TO THEM RSM2763X     342          I
          -CCG5
PROSEQ    *                                                                     RSM2763X     343          I
          -CCG5
PROSEQ    *       ENTRY  (X0) = CHANGE                                          RSM2763X     344          I
          -CCG5
PROSEQ                                                                          RSM2763X     345          I
          -CCG5
PROSEQ    AST     ENTRY.                                                        RSM2763X     346          I
          -CCG5
PROSEQ            PRINT  AST,(* AST CALLED,CHANGE =*Z7),(X0)                     RSM2763X     347          I
          -CCG5
PROSEQ            SA2    O.TEND                                                 RSM2763X     348          I
          -CCG1   -CCG5
PROSEQ            SA4    TN                                                     CCG1         238          I
          -CCG5
PROSEQ            SA2    FTAB+X4     FWA OF STATIC TABLES                       CCG1         239          I
          -CCG5
PROSEQ            SA4    =XCP.NFLS   FL-10                                      RSM2763X     349          I
          -CCG5
PROSEQ            IX7    X4+X0       NEW CP.NFLS = OLD CP.NFLS + CHANGE         RSM2763X     350          I
          -CCG5
PROSEQ            SA7    A4                                                     RSM2763X     351          I
          -CCG5
PROSEQ            IX1    X4-X2       WDS = CP.NFLS - O.TEND                     RSM2763X     352          I
          -CCG5
PROSEQ            IX3    X2+X0       TO = O.TEND + CHANGE                       RSM2763X     353          I
          -CCG5
PROSEQ            MOVE   X1,X2,X3                                               RSM2763X     354          I
          -CCG1   -CCG5
PROSEQ            ECHO   3,A=(O.TEND,O.GLT,O.API,O.LOOP,O.UDV,O.ENTR,O.EXT,_____ RSM2763X     355          I
          -RSM2763AX-CCG1    -CCG5
PROSEQ    ,LWAWORK)                                                             RSM2763X     356          I
          -RSM2763AX-CCG1    -CCG5
PROSEQ            SA4    =XO.LOOP                                               RSM2763AX      3           I
          -CCG1   -CCG5
PROSEQ    +       ZR     X4,*+1                                                 RSM2763AX      4           I
          -CCG1   -CCG5
PROSEQ            IX7    X4+X0                                                  RSM2763AX      5           I
          -CCG1   -CCG5
PROSEQ            SA7    A4                                                     RSM2763AX      6           I
          -CCG1   -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            ECHO    3,A=(O.TEND,O.GLT,O.API,O.UDV,O.ENTR,O.EXT,LWAWORK)          RSM2763AX      7          I
          -CCG1   -CCG5
PROSEQ            SA4     =X;A                                                         RSM2763X     357          I
          -CCG1   -CCG5
PROSEQ            IX7     X4+X0                                                        RSM2763X     358          I
          -CCG1   -CCG5
PROSEQ            SA7     A4                                                           RSM2763X     359          I
          -CCG1   -CCG5
PROSEQ            SA4     12B                                                          RSM2763X     360          I
          -CCG1   -CCG5
PROSEQ            IX7     X4+X0                                                        RSM2763X     361          I
          -CCG1   -CCG5
PROSEQ            SA7     A4                                                           RSM2763X     362          I
          -CCG1   -CCG5
PROSEQ            SA4     13B                                                          RSM2763X     363          I
          -CCG1   -CCG5
PROSEQ            IX7     X4+X0                                                        RSM2763X     364          I
          -CCG1   -CCG5
PROSEQ            SA7     A4                                                           RSM2763X     365          I
          -CCG1   -CCG5
PROSEQ            PRINT   AST,(*  NEW O.TEND =*Z7),(O.TEND)                            RSM2763X     366          I
          -CCG1   -CCG5
PROSEQ            SB2     X4                                                           CCG1         240          I
          -CCG5
PROSEQ            SB3     NTAB                                                         CCG1         241          I
          -CCG5
PROSEQ     AST1   SA4     FTAB+B2                                                      CCG1         242          I
          -CCG5
PROSEQ            IX7     X4+X0                                                        CCG1         243          I
          -CCG5
PROSEQ            SB2     B2+B1                                                        CCG1         244          I
          -CCG5
PROSEQ            SA7     A4                                                           CCG1         245          I
          -CCG5
PROSEQ            LT      B2,B3,AST1                                                   CCG1         246          I
          -CCG5
PROSEQ            MOVE    X1,X2,X3                                                     CCG1         247          I
          -CCG5
PROSEQ            EQ      AST                                                          RSM2763X     367          I
          -CCG5
PROSEQ    **      TOB - OPT=2 TABLE OVERFLOW PROCESSOR ( PHASE 2 )                     PROSEQ      2286          I
          -RSM2763X -CCG5
PROSEQ                                                                                 PROSEQ      2287          I
          -RSM2763X -CCG5
PROSEQ            ENTRY   TOB                                                          PROSEQ      2288          I
          -RSM2763X -CCG5
PROSEQ     TOB    SA2     =XMX.AVS                                                     PROSEQ      2289          I
          -RSM2763X -CCG5
PROSEQ            MX1     0                                                            PROSEQ      2290          I
          -RSM2763X -CCG5
PROSEQ            SB5     -B5                                                          PROSEQ      2291          I
          -RSM2763X -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            SX5    B5+200B     SD = SPACE NEEDED + 200B                          PROSEQ     2292        I
          -RSM2763X -CCG5
PROSEQ            IX6    X5-X2                                                         PROSEQ     2293        I
          -RSM2763X -CCG5
PROSEQ            MI     X6,TOB1     IF MX.AVS > SD                                    PROSEQ     2294        I
          -RSM2763X -CCG5
PROSEQ            SX5    X6+100B     SD = SD - MX.AVS + 100B                           PROSEQ     2295        I
          -RSM2763X -CCG5
PROSEQ            BX1    X5                                                            PROSEQ     2296        I
          -RSM2763X -CCG5
PROSEQ            CALL   DMB         CALL DMB( SD )                                    PROSEQ     2297        I
          -RSM2763X -CCG5
PROSEQ                                                                                PROSEQ     2298        I
          -RSM2763X -CCG5
PROSEQ    TOB1    NZ     X1,TOV      IF COULDN"T DUMP ENOUGH                           PROSEQ     2299        I
          -RSM2763X -CCG5
PROSEQ            SA2    L.BLK                                                         PROSEQ     2300        I
          -RSM2763X -CCG5
PROSEQ            SA3    =XM.EBL                                                       PROSEQ     2301        I
          -RSM2763X -CCG5
PROSEQ            IX6    X2-X5       L.BLK = L.BLK - SD                                PROSEQ     2302        I
          -RSM2763X -CCG5
PROSEQ            SA6    A2                                                            PROSEQ     2303        I
          -RSM2763X -CCG5
PROSEQ            IX7    X3-X5       M.EBL = M.EBL - SD                                PROSEQ     2304        I
          -RSM2763X -CCG5
PROSEQ            SA7    A3                                                            PROSEQ     2305        I
          -RSM2763X -CCG5
PROSEQ            CALL   CGB         COLLECT GARBAGE                                   PROSEQ     2306        I
          -RSM2763X -CCG5
PROSEQ            EQ     /TBLM/ATS9                                                    PROSEQ     2307        I
          -RSM2763X -CCG5
PROSEQ    PTC     SPACE  3,14                                                          PROSEQ     2308        I
          -CCG5
PROSEQ    .T      IFNE   TEST,0                                 * TEST MODE *         PROSEQ     2309        I
          -CCG5
PROSEQ                                                                                PROSEQ     2310        I
          -CCG5
PROSEQ    NAMTAB  BSS    0                                                             PROSEQ     2311        I
          -CCG5
PROSEQ    TABNAM  HERE                                                                 PROSEQ     2312        I
          -CCG5
PROSEQ                                                                                PROSEQ     2313        I
          -CCG5
PROSEQ    PTC=    ENTRY. **                                                            PROSEQ     2314        I
          -CCG5
PROSEQ            PRINT  ,(//*  -- COMPILER TABLE DUMP --*/)                           PROSEQ     2315        I
          -CCG5
PROSEQ            SX1    4           SET TO *EXECUTIVE IS /PTC=/*                      FCCT001      17        I
          -CCG5
PROSEQ            RJ     =XSOB       SET OUTPUT BIT FLAGS                              FCCT001      18        I
          -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            SX6    1                                                    PROSEQ     2316        I
          -CCG5
PROSEQ            SA6    PTCA        INITIALIZE COUNTER                       PROSEQ     2317        I
          -CCG5
PROSEQ                                                                        PROSEQ     2318        I
          -CCG5
PROSEQ    PTC1    SA1    PTCA                                                 PROSEQ     2319        I
          -CCG5
PROSEQ            SA2    FTAB+X1     GET FWA OF TABLE                         PROSEQ     2320        I
          -CCG5
PROSEQ            SA3    LTAB+X1     GET LENGTH OF TABLE                      PROSEQ     2321        I
          -CCG5
PROSEQ            SA5    NAMTAB+X1   GET NAME OF TABLE                        PROSEQ     2322        I
          -CCG5
PROSEQ            RJ     PMT                                                  PROSEQ     2323        I
          -CCG5
PROSEQ            SA1    PTCA                                                 PROSEQ     2324        I
          -CCG5
PROSEQ            SA2    TN                                                   PROSEQ     2325        I
          -CCG5
PROSEQ            SX6    X1+B1                                                PROSEQ     2326        I
          -CCG5
PROSEQ            SA6    A1          INCREMENT COUNTER                        PROSEQ     2327        I
          -CCG5
PROSEQ            IX7    X6-X2                                                PROSEQ     2328        I
          -CCG5
PROSEQ            MI     X7,PTC1     IF MORE TABLES                           PROSEQ     2329        I
          -CCG5
PROSEQ                                                                        PROSEQ     2330        I
          -CCG5
PROSEQ            SA1    =XOPT2                                               PROSEQ     2331        I
          -CCG1   -CCG5
PROSEQ            ZR     X1,PTC2     IF NOT OPT = 2                           PROSEQ     2332        I
          -CCG1   -CCG5
PROSEQ            SA1    =XCO.OPT                                             CCG1        248        I
          -CCG5
PROSEQ            PL     X1,PTC2     IF OPT " 2                               CCG1        249        I
          -CCG5
PROSEQ            ECHO   4,TBL=(BIT,SEQ)                                      PROSEQ     2333        I
          -CCG5
PROSEQ            SA2    =XO.TBL                                              PROSEQ     2334        I
          -CCG5
PROSEQ            SA3    =XL.TBL                                              PROSEQ     2335        I
          -CCG5
PROSEQ            SA5    =10H TBL                                             PROSEQ     2336        I
          -CCG5
PROSEQ            RJ     PMT                                                  PROSEQ     2337        I
          -CCG5
PROSEQ                                                                        PROSEQ     2338        I
          -CCG5
PROSEQ    PTC2    SA1    L.TXT                                                PROSEQ     2339        I
          -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            SB2    X1-8                                                  PROSEQ      2340          I
          -CCG5
PROSEQ            LE     B2,PTC3                                               PROSEQ      2341          I
          -CCG5
PROSEQ            SA1    O.TXT                                                 PROSEQ      2342          I
          -CCG5
PROSEQ            SA2    X1                                                    PROSEQ      2343          I
          -CCG5
PROSEQ            UX6    B2,X2                                                 PROSEQ      2344          I
          -CCG5
PROSEQ            LE     B2,PTC3     IF NO EXPONENT                            PROSEQ      2345          I
          -CCG5
PROSEQ            SNAPRL PTC                                                   PROSEQ      2346          I
          -CCG5
PROSEQ                                                                         PROSEQ      2347          I
          -CCG5
PROSEQ    PTC3    BSS    0                                                     PROSEQ      2348          I
          -CCG1   -CCG5
PROSEQ            IFEQ   CP#RM,0,1   WAIT UNTIL OUTPUT FINISHED SINCE *RPV*    FCC1126        5          I
          -CCG1   -CCG5
PROSEQ            RECALL =XF.OUT     UNCONDITIONALLY SETS THE COMPLETE BIT.    FCC1126        6          I
          -CCG1   -CCG5
PROSEQ            EQ     PTC=                                                  PROSEQ      2349          I
          -CCG1   -CCG5
PROSEQ    PTC3    RECALL =XF.OUT                                               CCG1         250          I
          -CCG5
PROSEQ            EQ     PTC=                                                  CCG1         251          I
          -CCG5
PROSEQ                                                                         PROSEQ      2350          I
          -CCG5
PROSEQ    PTCA    BSS    1                                                     PROSEQ      2351          I
          -CCG5
PROSEQ                                                                         PROSEQ      2352          I
          -CCG5
PROSEQ    PMT     ROUTINE                                                      PROSEQ      2353          I
          -CCG5
PROSEQ            ZR     X3,PMT      IF LEN = 0                                PROSEQ      2354          I
          -CCG5
PROSEQ            SB3    X2                                                    PROSEQ      2355          I
          -FEAT155F -CCG5
PROSEQ            SX7    X3                                                    PROSEQ      2356          I
          -FEAT155F -CCG5
PROSEQ            PRINT  ,(/*  TABLE -*A10*  FWA,LEN =*2Z6/),(X5,X2,X3)        PROSEQ      2357          I
          -CCG5
PROSEQ            CALL   DMPC=                                                 PROSEQ      2358          I
          -FEAT155F -CCG5
PROSEQ            IX3    X2+X3                                                 FEAT155F     258          I
          -CCG5
PROSEQ            MX4    0                                                     FEAT155F     259          I
          -CCG5
PROSEQ            RJ     =XDCM=      DUMP CENTRAL MEMORY                       FEAT155F     260          I
          -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            EQ     PMT                                                    PROSEQ      2359          I
          -CCG5
PROSEQ                                                                          PROSEQ      2360          I
          -CCG5
PROSEQ    .T      ENDIF                                   * TEST MODE *         PROSEQ      2361          I
          -CCG5
PROSEQ    ATS     TITLE  TABLE MANAGER SUBROUTINES                              PROSEQ      2362          I
          -CCG5
PROSEQ            QUAL   TBLM                                                   PROSEQ      2363          I
          -CCG5
PROSEQ                                                                          PROSEQ      2364          I
          -CCG5
PROSEQ    **      ATS - ALLOCATE TABLE SPACE.                                   PROSEQ      2365          I
          -CCG5
PROSEQ    *                                                                     PROSEQ      2366          I
          -CCG5
PROSEQ    *       ENTRY  (A0) = TABLE INDEX.                                    PROSEQ      2367          I
          -CCG5
PROSEQ    *              (X1) = CHANGE (+ OR -) TO TABLE SIZE.                  PROSEQ      2368          I
          -CCG5
PROSEQ    *                                                                     PROSEQ      2369          I
          -CCG5
PROSEQ    *       EXIT   (X1) = CHANGE.                                         PROSEQ      2370          I
          -CCG5
PROSEQ    *              (X2) = FWA OF TABLE.                                   PROSEQ      2371          I
          -CCG5
PROSEQ    *              (X3) = NEW LENGTH OF TABLE.                            PROSEQ      2372          I
          -CCG5
PROSEQ    *              (B6) = PREVIOUS TABLE LENGTH                           PROSEQ      2373          I
          -CCG5
PROSEQ    *                                                                     PROSEQ      2374          I
          -CCG5
PROSEQ    *       USES   X - 0, 1, 2, 3, 4, 5, 6, 7.                            PROSEQ      2375          I
          -CCG5
PROSEQ    *              B - 2, 3, 4, 5, 7                                      PROSEQ      2376          I
          -CCG5
PROSEQ    *              A - 1, 2, 3, 4, 6, 7                                   PROSEQ      2377          I
          -CCG5
PROSEQ    *       RESTORES X0, X5.                                              PROSEQ      2378          I
          -CCG5
PROSEQ    *       CALLS  AMU, MVE.                                              PROSEQ      2379          I
          -CCG5
PROSEQ                                                                          PROSEQ      2380          I
          -CCG5
PROSEQ    ATSX    SA2    FTAB+A0     (X2) = FWA TABLE                           PROSEQ      2381          I
          -CCG5
PROSEQ            SA3    LTAB+A0     (X3) = NEW LENGTH                          PROSEQ      2382          I
          -CCG5
PROSEQ                                                                          PROSEQ      2383          I
          -CCG5
PROSEQ    ATS     PS                 ENTRY/EXIT                                 PROSEQ      2384          I
          -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ    ATS1     SA2    FTAB+A0     CURRENT FWA                                  PROSEQ    2385    I
          -CCG5
PROSEQ             SA3    LTAB+A0     CURRENT LENGTH                               PROSEQ    2386    I
          -CCG5
PROSEQ             SA4    A2+B1       NEXT TABLE FWA                               PROSEQ    2387    I
          -CCG5
PROSEQ             IX6    X1+X3       NEW LENGTH                                   PROSEQ    2388    I
          -CCG5
PROSEQ             SB6    X3          (B6) = PREVIOUS TABLE LENGTH                 PROSEQ    2389    I
          -CCG5
PROSEQ             IX7    X4-X2       ROOM BETWEEN TABLES                          PROSEQ    2390    I
          -CCG5
PROSEQ             SA6    A3          SET NEW LENGTH                               PROSEQ    2391    I
          -CCG5
PROSEQ             IX7    X7-X6                                                    PROSEQ    2392    I
          -CCG5
PROSEQ             PL     X7,ATSX     RETURN IF ROOM FOR CHANGE                    PROSEQ    2393    I
          -CCG5
PROSEQ             SX7    X3          SET PREVIOUS LENGTH                          PROSEQ    2394    I
          -CCG5
PROSEQ                                                                             PROSEQ    2395    I
          -CCG5
PROSEQ    **       INITIALIZE FOR TABLE MOVE.                                      PROSEQ    2396    I
          -CCG5
PROSEQ                                                                             PROSEQ    2397    I
          -CCG5
PROSEQ             SA2    TN          (B2) = NUMBER OF TABLES                      PROSEQ    2398    I
          -CCG5
PROSEQ             BX6    X0          SAVE (X0)                                    PROSEQ    2399    I
          -CCG5
PROSEQ             SA7    A3                                                       PROSEQ    2400    I
          -CCG5
PROSEQ             SA7    PL          PL = PREVIOUS LENGTH                         PROSEQ    2401    I
          -CCG5
PROSEQ             SB2    X2                                                       PROSEQ    2402    I
          -CCG5
PROSEQ             LX7    X5          SAVE (X5)                                    PROSEQ    2403    I
          -CCG5
PROSEQ             SA6    A2+B1                                                    PROSEQ    2404    I
          -CCG5
PROSEQ             SA7    A6+B1                                                    PROSEQ    2405    I
          -CCG5
PROSEQ             BX6    X1          SAVE (X1)                                    PROSEQ    2406    I
          -CCG5
PROSEQ             SA6    A7+B1                                                    PROSEQ    2407    I
          -CCG5
PROSEQ                                                                             PROSEQ    2408    I
          -CCG5
PROSEQ    **       COMPUTE REMAINING TABLE SPACE.                                  PROSEQ    2409    I
          -CCG5
PROSEQ                                                                             PROSEQ    2410    I
          -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            SB3    B2-B1                                                PROSEQ     2411        I
          -CCG5
PROSEQ            BX4    X1          INCLUDE INCREASE                         PROSEQ     2412        I
          -CCG5
PROSEQ    ATS2    SB3    B3-B1       ACCUMULATE ASSIGNED TABLE LENGTHS        PROSEQ     2413        I
          -CCG5
PROSEQ            SA3    LTAB+B3                                              PROSEQ     2414        I
          -CCG5
PROSEQ            IX4    X4+X3                                                PROSEQ     2415        I
          -CCG5
PROSEQ            NZ     B3,ATS2     LOOP FOR ALL TABLES                      PROSEQ     2416        I
          -CCG5
PROSEQ            SA2    LM          SET AVAILABLE LENGTH                     PROSEQ     2417        I
          -CCG5
PROSEQ            SA3    FTAB-1+B2                                            PROSEQ     2418        I
          -CCG5
PROSEQ            IX6    X3-X2                                                PROSEQ     2419        I
          -CCG5
PROSEQ            SB4    X4          (B4) = TOTAL ASSIGNED LENGTH             PROSEQ     2420        I
          -CCG5
PROSEQ            IX7    X6-X4                                                PROSEQ     2421        I
          -CCG5
PROSEQ            SB5    X7          (B5) = REMAINING SPACE                   PROSEQ     2422        I
          -CCG5
PROSEQ            NG     X7,ATS8     IF NO REMAINING SPACE                    PROSEQ     2423        I
          -CCG5
PROSEQ                                                                        PROSEQ     2424        I
          -CCG5
PROSEQ    **      MOVE ALL TABLES DOWN.                                       PROSEQ     2425        I
          -CCG5
PROSEQ                                                                        PROSEQ     2426        I
          -CCG5
PROSEQ            SA1    LM          LOW MEMORY LIMIT                         PROSEQ     2427        I
          -CCG5
PROSEQ            SB3    B1                                                   PROSEQ     2428        I
          -CCG5
PROSEQ            LX0    X1                                                   PROSEQ     2429        I
          -CCG5
PROSEQ    ATS3    SA2    FTAB-1+B3   ORIGIN = PREVIOUS FWA                    PROSEQ     2430        I
          -CCG5
PROSEQ            SA1    LTAB-1+B3   WORD COUNT = LENGTH                      PROSEQ     2431        I
          -CCG5
PROSEQ            BX3    X0          DESTINATION = NEW FWA                    PROSEQ     2432        I
          -CCG5
PROSEQ            IX0    X0+X1       NEXT FWA                                 PROSEQ     2433        I
          -CCG5
PROSEQ            SX7    X3          SET NEW FWA                              PROSEQ     2434        I
          -CCG5
PROSEQ            BX4    X2-X3       MOVE DIFFERENTIAL                        PROSEQ     2435        I
          -CCG5
PROSEQ            SA7    A2                                                   PROSEQ     2436        I
          -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            SB3    B3+B1       COUNT TABLE                                          PROSEQ     2437        I
          -CCG5
PROSEQ            ZR     X4,ATS4     IF NO MOVE REQUIRED                                  PROSEQ     2438        I
          -CCG5
PROSEQ            RJ     MVE=                                                             PROSEQ     2439        I
          -CCG5
PROSEQ    ATS4    NE     B3,B2,ATS3  LOOP FOR ALL TABLES                                  PROSEQ     2440        I
          -CCG5
PROSEQ            SA1    TN+3        INCREMENT SIZE OF REQUESTED TABLE                    PROSEQ     2441        I
          -CCG5
PROSEQ            SA2    LTAB+A0                                                          PROSEQ     2442        I
          -CCG5
PROSEQ            IX6    X2+X1                                                            PROSEQ     2443        I
          -CCG5
PROSEQ            SA6    A2                                                               PROSEQ     2444        I
          -CCG5
PROSEQ                                                                                    PROSEQ     2445        I
          -CCG5
PROSEQ    **      REALLOCATE TABLES.                                                      PROSEQ     2446        I
          -CCG5
PROSEQ    *       ALLOCATE INTERSPACE.  (LA/2N) + ((TL*LA)/2)                             PROSEQ     2447        I
          -CCG5
PROSEQ                                                                                    PROSEQ     2448        I
          -CCG5
PROSEQ            SA1    FTAB-1+B2   (X0) = LWA+1 LAST TABLE                              PROSEQ     2449        I
          -CCG5
PROSEQ            SB3    B2-B1                                                            PROSEQ     2450        I
          -CCG5
PROSEQ            BX0    X1                                                               PROSEQ     2451        I
          -CCG5
PROSEQ    ATS4A   SB6    2           LOWER LIMIT = 2 ( FIRST TABLE INACTIVE )             PROSEQ     2452        I
          -CCG5
PROSEQ            SB2    B2-1        N = N - 1                                            PROSEQ     2453        I
          -CCG5
PROSEQ    ATS5    SB3    B3-B1       I = I + 1                                            PROSEQ     2454        I
          -CCG5
PROSEQ            SA2    FTAB+B3     FWA(I)                                               PROSEQ     2455        I
          -CCG5
PROSEQ            SX4    B5          LA (LENGTH AVAILABLE)                                PROSEQ     2456        I
          -CCG5
PROSEQ            SA1    LTAB+B3     TL (LENGTH OF TABLE)                                 PROSEQ     2457        I
          -CCG5
PROSEQ            SX3    B2-B1       N  (NUMBER OF TABLES)                                PROSEQ     2458        I
          -CCG5
PROSEQ            AX5    X4,B1       LA/2                                                 PROSEQ     2459        I
          -CCG5
PROSEQ            IX6    X4/X3       L1 = LA/N                                            PROSEQ     2460        I
          -CCG5
PROSEQ            ZR     B4,ATS6     IF NO TABLES ASSIGNED, L = L2                        PROSEQ     2461        I
          -CCG5
PROSEQ            SX3    B4          AL (TOTAL ASSIGNED LENGTH)                           PROSEQ     2462        I
          -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ              IX7    X5*X1       (LA/2)*TL                                PROSEQ     2463        I
          -CCG5
PROSEQ              AX6    1           L1 = LA/2N                               PROSEQ     2464        I
          -CCG5
PROSEQ              IX7    X7/X3       L2 = (TL*LA)/2                           PROSEQ     2465        I
          -CCG5
PROSEQ              IX6    X6+X7       L = L1+L2                                PROSEQ     2466        I
          -CCG5
PROSEQ              SX1    X1                                                   PROSEQ     2467        I
          -CCG5
PROSEQ     ATS6     IX6    X0-X6       FWA(I+1)-L                               PROSEQ     2468        I
          -CCG5
PROSEQ              IX7    X6-X1       FWA(I) = FWA(I+1)-L-TL                   PROSEQ     2469        I
          -CCG5
PROSEQ              BX3    X7          DESTINATION = FWA(I)                     PROSEQ     2470        I
          -CCG5
PROSEQ              SA7    A2                                                   PROSEQ     2471        I
          -CCG5
PROSEQ              LX0    X7          FWA(I+1) = FWA(I)                        PROSEQ     2472        I
          -CCG5
PROSEQ              BX4    X2-X3       CHECK MOVE DIFFERENTIAL                  PROSEQ     2473        I
          -CCG5
PROSEQ              ZR     X4,ATS7     IF NO MOVE REQUIRED                      PROSEQ     2474        I
          -CCG5
PROSEQ              RJ     MVE=        MOVE TABLE                               PROSEQ     2475        I
          -CCG5
PROSEQ     ATS7     GT     B3,B6,ATS5  LOOP                                     PROSEQ     2476        I
          -CCG5
PROSEQ              RJ     AMU         ACCUMULATE MEMORY USED                   PROSEQ     2477        I
          -CCG1     -CCG5
PROSEQ                                                                          PROSEQ     2478        I
          -CCG5
PROSEQ    **        RESTORE REGISTERS.                                           PROSEQ     2479        I
          -CCG5
PROSEQ                                                                          PROSEQ     2480        I
          -CCG5
PROSEQ              SA1    ATSA        COUNT MOVES                              PROSEQ     2481        I
          -CCG5
PROSEQ              SX6    X1+B1                                                PROSEQ     2482        I
          -CCG5
PROSEQ              SA6    A1                                                   PROSEQ     2483        I
          -CCG5
PROSEQ              SA2    TN+1        RESTORE (X0)                             PROSEQ     2484        I
          -CCG5
PROSEQ              SA3    A2+B1       RESTORE (X5)                             PROSEQ     2485        I
          -CCG5
PROSEQ              BX0    X2                                                   PROSEQ     2486        I
          -CCG5
PROSEQ              SA4    PL                                                   PROSEQ     2487        I
          -CCG5
PROSEQ              LX5    X3                                                   PROSEQ     2488        I
          -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            SA1    A3+B1       RESTORE (X1)                                  PROSEQ    2489        I
          -CCG5
PROSEQ            SB6    X4          (B6) = PREVIOUS LENGTH OF TABLE               PROSEQ    2490        I
          -CCG5
PROSEQ            EQ     ATSX        RETURN                                        PROSEQ    2491        I
          -CCG5
PROSEQ                                                                             PROSEQ    2492        I
          -CCG5
PROSEQ     ATS8   SA3    TO          PROCESS TABLE OVERFLOW                        PROSEQ    2493        I
          -CCG5
PROSEQ            SX6    A0                                                        PROSEQ    2494        I
          -CCG5
PROSEQ            SB2    X3                                                        PROSEQ    2495        I
          -CCG5
PROSEQ            SA6    TN+4        (A0)                                          PROSEQ    2496        I
          -CCG5
PROSEQ            JP     B2          (B5) = - WORDS NEEDED                         PROSEQ    2497        I
          -CCG5
PROSEQ                                                                             PROSEQ    2498        I
          -CCG5
PROSEQ    *       RETURN FROM USER TABLE OVERFLOW PROCESSOR ( SUCCESS )            PROSEQ    2499        I
          -CCG5
PROSEQ                                                                             PROSEQ    2500        I
          -CCG5
PROSEQ     ATS9   SA1    TN+3        X1                                            PROSEQ    2501        I
          -CCG5
PROSEQ            SA4    A1-B1       X5                                            PROSEQ    2502        I
          -CCG5
PROSEQ            BX5    X4                                                        PROSEQ    2503        I
          -CCG5
PROSEQ            SA3    A1+B1       A0                                            PROSEQ    2504        I
          -CCG5
PROSEQ            SA0    X3                                                        PROSEQ    2505        I
          -CCG5
PROSEQ            SA4    A4-B1       X0                                            PROSEQ    2506        I
          -CCG5
PROSEQ            BX0    X4                                                        PROSEQ    2507        I
          -CCG5
PROSEQ            SA2    A4-B1       TN                                            PROSEQ    2508        I
          -CCG5
PROSEQ            SB2    X2          (B2) = TN                                     PROSEQ    2509        I
          -CCG5
PROSEQ            SB3    B2-B1                                                     PROSEQ    2510        I
          -CCG5
PROSEQ            BX4    X1                                                        PROSEQ    2511        I
          -CCG5
PROSEQ            EQ     ATS2                                                      PROSEQ    2512        I
          -CCG5
PROSEQ                                                                             PROSEQ    2513        I
          -CCG5
PROSEQ     ATSA   CON    0           COUNT OF TABLE MOVES                          PROSEQ    2514        I
          -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ    AFT      SPACE  3,14                                                    PROSEQ      2515          I
          -CCG5
PROSEQ    **       AFT - ACTIVATE FIRST TABLE                                      PROSEQ      2516          I
          -CCG5
PROSEQ                                                                             PROSEQ      2517          I
          -CCG5
PROSEQ    AFT      PS                                                              PROSEQ      2518          I
          -CCG5
PROSEQ             SA1    AFTA                                                     PROSEQ      2519          I
          -CCG5
PROSEQ             SA2    LM                                                       PROSEQ      2520          I
          -CCG5
PROSEQ             SA3    ATS4A                                                    PROSEQ      2521          I
          -CCG5
PROSEQ             BX6    X1                                                       PROSEQ      2522          I
          -CCG5
PROSEQ             LX7    X2                                                       PROSEQ      2523          I
          -CCG5
PROSEQ             SA6    ATS4A       STORE PLUG                                   PROSEQ      2524          I
          -CCG5
PROSEQ             SA7    FTAB                                                     PROSEQ      2525          I
          -CCG5
PROSEQ             BX7    X3                                                       PROSEQ      2526          I
          -CCG5
PROSEQ             SA7    A1                                                       PROSEQ      2527          I
          -CCG5
PROSEQ             EQ     AFT                                                      PROSEQ      2528          I
          -CCG5
PROSEQ                                                                             PROSEQ      2529          I
          -CCG5
PROSEQ    AFTA     SB6    1                                                        PROSEQ      2530          I
          -CCG5
PROSEQ             SB0    0                                                        PROSEQ      2531          I
          -CCG5
PROSEQ    ADW      EJECT              ADW                                           PROSEQ      2532          I
          -CCG5
PROSEQ    **       ADW - ADD WORD TO TABLE.                                         PROSEQ      2533          I
          -CCG5
PROSEQ    *        ENTRY  (A0) = TABLE NUMBER.                                      PROSEQ      2534          I
          -CCG5
PROSEQ    *               (X1) = WORD.                                              PROSEQ      2535          I
          -CCG5
PROSEQ    *        EXIT   (X1) = WORD.                                              PROSEQ      2536          I
          -CCG5
PROSEQ    *               (X6) = WORD.                                              PROSEQ      2537          I
          -CCG5
PROSEQ    *               (A6) = ADDRESS OF WORD.                                   PROSEQ      2538          I
          -CCG5
PROSEQ    *               (X2) = FWA TABLE.                                         PROSEQ      2539          I
          -CCG5
PROSEQ    *               (X3) = LENGTH OF TABLE.                                   PROSEQ      2540          I
          -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ   *        USES   X - 1, 6, 7.                                          PROSEQ    2541        I
         -CCG5
PROSEQ   *               B - NONE.                                             PROSEQ    2542        I
         -CCG5
PROSEQ   *               A - 1, 6.                                             PROSEQ    2543        I
         -CCG5
PROSEQ   *        CALLS  ATS.                                                  PROSEQ    2544        I
         -CCG5
PROSEQ                                                                         PROSEQ    2545        I
         -CCG5
PROSEQ    ADW1    BX7    X3                                                    PROSEQ    2546        I
         -CCG5
PROSEQ            IX4    X2+X3                                                 PROSEQ    2547        I
         -CCG5
PROSEQ            SA7    A3+         UPDATE LENGTH                             PROSEQ    2548        I
         -CCG5
PROSEQ            SB0    0                                                     PROSEQ    2549        I
         -CCG5
PROSEQ            SA6    X4-1        STORE WORD                                PROSEQ    2550        I
         -CCG5
PROSEQ    ADW     PS                                                           PROSEQ    2551        I
         -CCG5
PROSEQ            SA2    FTAB+A0     FWA                                       PROSEQ    2552        I
         -CCG5
PROSEQ            SA3    LTAB+A0                                               PROSEQ    2553        I
         -CCG5
PROSEQ            SA4    A2+B1       NEXT TABLE FWA                            PROSEQ    2554        I
         -CCG5
PROSEQ            BX6    X1                                                    PROSEQ    2555        I
         -CCG5
PROSEQ            IX7    X4-X2       ROOM BETWEEN TABLES                       PROSEQ    2556        I
         -CCG5
PROSEQ            SX3    X3+B1                                                 PROSEQ    2557        I
         -CCG5
PROSEQ            IX4    X7-X3                                                 PROSEQ    2558        I
         -CCG5
PROSEQ            PL     X4,ADW1     IF ROOM TO ADD A WORD                     PROSEQ    2559        I
         -CCG5
PROSEQ            SX1    B1                                                    PROSEQ    2560        I
         -CCG5
PROSEQ            SA6    ADWA                                                  PROSEQ    2561        I
         -CCG5
PROSEQ            RJ     ATS         ALLOCATE SPACE                            PROSEQ    2562        I
         -CCG5
PROSEQ            SA1    ADWA                                                  PROSEQ    2563        I
         -CCG5
PROSEQ            IX4    X2+X3                                                 PROSEQ    2564        I
         -CCG5
PROSEQ            BX6    X1                                                    PROSEQ    2565        I
         -CCG5
PROSEQ            SA6    X4-1                                                  PROSEQ    2566        I
         -CCG5

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            EQ     ADW                                                PROSEQ     2567        I
          -CCG5
PROSEQ                                                                      PROSEQ     2568        I
          -CCG5
PROSEQ    ADWA    BSS    1                                                  PROSEQ     2569        I
          -CCG5
PROSEQ    AMU     SPACE  3                                                  PROSEQ     2570        I
          -CCG1   -CCG5
PROSEQ    **      AMU - ACCUMULATE MEMORY USED.                             PROSEQ     2571        I
          -CCG1   -CCG5
PROSEQ    *       ENTRY  NONE.                                              PROSEQ     2572        I
          -CCG1   -CCG5
PROSEQ    *       EXIT   MU = MAX(MU,CURRENT ASSIGNED LENGTH)               PROSEQ     2573        I
          -CCG1   -CCG5
PROSEQ    *       USES   X - 1, 2, 3, 6.                                    PROSEQ     2574        I
          -CCG1   -CCG5
PROSEQ    *              B - 2.                                             PROSEQ     2575        I
          -CCG1   -CCG5
PROSEQ    *              A - 1, 2, 6.                                       PROSEQ     2576        I
          -CCG1   -CCG5
PROSEQ    *       CALLS  NONE.                                              PROSEQ     2577        I
          -CCG1   -CCG5
PROSEQ                                                                      PROSEQ     2578        I
          -CCG1   -CCG5
PROSEQ    AMU1    IX6    X6+X2       ACCUMULATE LENGTH                      PROSEQ     2579        I
          -CCG1   -CCG5
PROSEQ            SB2    B2-B1                                              PROSEQ     2580        I
          -CCG1   -CCG5
PROSEQ            SA2    A2+1        NEXT TABLE                             PROSEQ     2581        I
          -CCG1   -CCG5
PROSEQ            NE     B2,B1,AMU1  LOOP FOR ALL TABLES                    PROSEQ     2582        I
          -CCG1   -CCG5
PROSEQ            SA1    MU          SET MAX LENGTH                         PROSEQ     2583        I
          -CCG1   -CCG5
PROSEQ            IX3    X6-X1                                              PROSEQ     2584        I
          -CCG1   -CCG5
PROSEQ            NG     X3,AMU                                             PROSEQ     2585        I
          -CCG1   -CCG5
PROSEQ            SA6    A1                                                 PROSEQ     2586        I
          -CCG1   -CCG5
PROSEQ                                                                      PROSEQ     2587        I
          -CCG1   -CCG5
PROSEQ    AMU     PS                 ENTRY/EXIT                             PROSEQ     2588        I
          -CCG1   -CCG5
PROSEQ            SA1    TN          (B2) = NUMBER OF TABLES                PROSEQ     2589        I
          -CCG1   -CCG5
PROSEQ            MX6    0           CLEAR ACCUMULATION                     PROSEQ     2590        I
          -CCG1   -CCG5
PROSEQ            SB2    X1                                                 PROSEQ     2591        I
          -CCG1   -CCG5
PROSEQ            SA2    LTAB        FIRST TABLE                            PROSEQ     2592        I
          -CCG1   -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            EQ     AMU1                                                PROSEQ     2593        I
          -CCG1   -CCG5
PROSEQ     MTU    SPACE  3                                                   PROSEQ     2594        I
          -CCG5
PROSEQ    **      MTU - MOVE TABLES UP.                                      PROSEQ     2595        I
          -CCG5
PROSEQ    *       ENTRY  NONE.                                               PROSEQ     2596        I
          -CCG5
PROSEQ    *       EXIT   NONE.                                               PROSEQ     2597        I
          -CCG5
PROSEQ    *       USES   X - 0, 1, 2, 3, 7.                                  PROSEQ     2598        I
          -CCG5
PROSEQ    *              B - 3.                                              PROSEQ     2599        I
          -CCG5
PROSEQ    *              A - 1, 2, 7.                                        PROSEQ     2600        I
          -CCG5
PROSEQ    *       CALLS  MVE.                                                PROSEQ     2601        I
          -CCG5
PROSEQ                                                                       PROSEQ     2602        I
          -CCG5
PROSEQ     MTU1   SB3    B3-B1       DECREMENT TABLE COUNT                   PROSEQ     2603        I
          -CCG5
PROSEQ            SA2    FTAB+B3                                             PROSEQ     2604        I
          -CCG5
PROSEQ            SA1    LTAB+B3                                             PROSEQ     2605        I
          -CCG5
PROSEQ            IX7    X0-X1       NEW FWA = L - LENGTH                    PROSEQ     2606        I
          -CCG5
PROSEQ            LX0    X7          L = NEW FWA                             PROSEQ     2607        I
          -CCG5
PROSEQ            SA7    A2                                                  PROSEQ     2608        I
          -CCG5
PROSEQ            BX3    X0          MOVE TABLE                              PROSEQ     2609        I
          -CCG5
PROSEQ            ZR     X1,MTU2     IF LEN = 0                              PROSEQ     2610        I
          -CCG5
PROSEQ            IX4    X2-X3                                               PROSEQ     2611        I
          -CCG5
PROSEQ            ZR     X4,MTU2                                             PROSEQ     2612        I
          -CCG5
PROSEQ            MOVE   X1,X2,X3                                            PROSEQ     2613        I
          -CCG5
PROSEQ     MTU2   NZ     B3,MTU1                                             PROSEQ     2614        I
          -CCG5
PROSEQ                                                                       PROSEQ     2615        I
          -CCG5
PROSEQ     MTU    PS                 ENTRY/EXIT                              PROSEQ     2616        I
          -CCG5
PROSEQ            SB3    NTAB-1      (B3) = NUMBER OF TABLES - 1             PROSEQ     2617        I
          -CCG5
PROSEQ            SA1    FTAB+B3     (X0) = LWA+1 OF ALL TABLES (L)          PROSEQ     2618        I
          -CCG5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

```
PROSEQ            BX0    X1                                          PROSEQ     2619        I
          -CCG5
PROSEQ            EQ     MTU1                                        PROSEQ     2620        I
          -CCG5
PROSEQ            SPACE  2                                           PROSEQ     2621        I
          -CCG5
PROSEQ            QUAL                                               PROSEQ     2622        I
          -CCG5
PROSEQ            ENTRY  ADW=,ATS=,AMU=,MTU=                         PROSEQ     2623        I
          -CCG1   -CCG5
PROSEQ            ENTRY  AFT=                                        PROSEQ     2624        I
          -CCG1   -CCG5
PROSEQ    AFT=    EQU    /TBLM/AFT                                   PROSEQ     2625        I
          -CCG1   -CCG5
PROSEQ    ADW=    EQU    /TBLM/ADW                                   PROSEQ     2626        I
          -CCG1   -CCG5
PROSEQ    AMU=    EQU    /TBLM/AMU                                   PROSEQ     2627        I
          -CCG1   -CCG5
PROSEQ    ATS=    EQU    /TBLM/ATS                                   PROSEQ     2628        I
          -CCG1   -CCG5
PROSEQ    MTU=    EQU    /TBLM/MTU                                   PROSEQ     2629        I
          -CCG1   -CCG5
PROSEQ                                                               CCG1        252        I
          -CCG5
PROSEQ            ECHO   2,Z=(AFT,ADW,ATS,MTU)                       CCG1        253        I
          -CCG5
PROSEQ    Z=      EQU    /TBLM/Z                                     CCG1        254        I
          -CCG5
PROSEQ            ENTRY  Z=                                          CCG1        255        I
          -CCG5
PROSEQ            ECHO   2,Z=(FTAB,LTAB)                             CCG1        256        I
          -CCG5
PROSEQ    Z=      EQU    Z                                           CCG1        257        I
          -CCG5
PROSEQ            ENTRY  Z=                                          CCG1        258        I
          -CCG5
PROSEQ                                                               PROSEQ     2630     A
PROSEQ            END                                                PROSEQ     2631     A
```

SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - PROSEQ

IDENTIFIER	TOTAL	ACTIVE
PROSEQ	2631	1752
FCC0504	1	0
FCC0508	16	16
FCC0586	18	18
FCC0641	2	2
FCC0643	33	32
FCC0645	8	7
FCC0882	3	2
FCC0912	10	10

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PROSEQ

FEAT155A	1	0
FEAT155AB	4	1
FEAT155F	4	0
FCC1126	2	0
FCC1239	5	5
FCCT001	2	0
FCC1353	18	17
RSM2763X	198	0
RSM2763AX	5	0
CCG1	107	18
CCG5	67	30
FCC1596X	3	3
CCG7	4	0
FCC1677X	5	5
CCGNC	76	74
CCG11	1	1
CCG12	1	1
CCG13	6	6
FCC1837X	26	26
CCG16	3	3
CCG18	34	32
FCC1918	15	15
CCG27	1	1
CCG35	58	33
F2950CC	5	5
CCGA014	2	2
CCGA040	29	29
CCGA044	115	93
CCGA052	10	10
CCG0085	1	1
CCG0088	12	12
CCGA099	4	3
CCGA100	1	1

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ      *DECK     SQZ                                                    SQZ          1     A
SQZ                IDENT  SQZ                                             SQZ          2     A
SQZ       SBB      TITLE  SQZ - REDUNDANT OPERATION ELIMINATION           SQZ          3     A
SQZ      *CALL     SSTCALL                                                SQZ          4     A
SQZ                PASS2TM                                                SQZ          5          I
          -CCG1
SQZ       B=SQZ    RPVDEF                                                 SQZ          6     A
SQZ                                                                       SQZ          7     A
SQZ      *IF       DEF,SQZ,1                                              SQZ          8     A
SQZ                TRACER (INITX,EDD,SQZ.X)                               SQZ          9     A
SQZ                TABLES TXT,RND                                         SQZ         10          I
          -CCG1
SQZ                                                                       SQZ         11          I
          -CCG1
SQZ                EXT    F.RDT,BRN,BIR                                   SQZ         12          I
          -CCG1

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                ECHO   1,Z=(CUT,CVT,RND,TET,TXT)                          CCG1        259          I
          -CCGNC
SQZ                EXT    O.Z,L.Z                                            CCG1        260          I
          -CCGNC
SQZ                TABLES (CUT,CVT,MOD,RND,TET,TXT)                          CCGNC       746     A
SQZ                EXT#   (N.GT,L.HB)                                        CCGNC       747     A
SQZ                                                                         SQZ          13     A
SQZ       **       SQZ - SQUEEZE INSTRUCTION SEQUENCE TO -                  SQZ          14     A
SQZ       *                ELIMINATE REDUNDANT AND UNUSED INSTRUCTIONS     SQZ          15     A
SQZ       *                INSTRUCTION SIMPLIFICATION ( ALGEBRAIC IDENTITIES )  SQZ      16     A
SQZ       *                COMPILE TIME EVALUATION OF CONSTANT EXPRESSIONS  SQZ          17     A
SQZ       *                GLOBAL TEMPORARY SQUEEZING ( IT."S )             SQZ          18     A
SQZ       *                                                                 SQZ          19     A
SQZ       *        ENTRY  *TXT* TABLE HOLDS AN ARBITRARY COLLECTION OF BLOCKS  SQZ       20     A
SQZ       *        TERMINATED BY AN *EOQ*.  R-NUMBERS MAY NOT "ROLL OVER".  SQZ          21     A
SQZ       *                                                                 SQZ          22     A
SQZ       *        EXIT   BLOCK SQUEEZED, REDUNDANT AND USELESS INSTRUCTIONS  SQZ       23     A
SQZ       *        REMOVED. PRECEDENCE BITS ( PS,PRS,RJRS ) SET AND USES COUNTS  SQZ    24     A
SQZ       *        INSTALLED, LINK WORDS CLEARED.                           SQZ          25          I
          -FCC0716
SQZ       *        INSTALLED, LINK WORDS CLEARED AND *USI* BIT SET FOR REDUCED  FCC0716  10     A
SQZ       *        *IAZ/ISZ* INSTRUCTIONS TO PREVENT OVERSQUEEZING IN OPT=2 .  FCC0716  11     A
SQZ       *                                                                 SQZ          26     A
SQZ       *        S.I. JASIK - JAN 1973                                    SQZ          27     A
SQZ                SPACE  2                                                 SQZ          28     A
SQZ                ENTRY  SBB                                               SQZ          29          I
          -CCGNC
SQZ        STLDS   ENTRY. 0,#         ST/LD SQZ FLAG, "0 IF ST/LD SQZ DONE  CCGNC       748     A
SQZ        SUVA    ENTRY. 0,#         SQUEEZED USE *BV* ADDRESS ( *AUV* CALL )  CCGNC   749     A
SQZ        LCM#G   ENTRY. 0           = 1S( D.L2++59-D.STP ) IF LCM=G ON CC  CCG21       68     A
SQZ                                                                         CCGNC       750     A
SQZ                QUAL   SQUEEZE                                           SQZ          30     A
SQZ                SPACE  2                                                 SQZ          31     A
SQZ        CLEAR   MACRO              CLEAR LINK CHAINS                     SQZ          32     A
SQZ                MX6    0                                                 SQZ          33     A
SQZ                SA6    MRC                                               SQZ          34     A
SQZ                DUP    N.CHAIN-1,1                                       SQZ          35     A
SQZ                SA6    A6+B1                                             SQZ          36     A
SQZ                ENDM                                                     SQZ          37     A
SQZ        LINK    SPACE  2                                                 SQZ          38     A
SQZ       **       I. - LINK INFORMATION WORD FORMAT DURING SQUEEZING       SQZ          39     A
SQZ                                                                         SQZ          40     A
SQZ                DESCRIBE I.,60                                           SQZ          41     A
SQZ        JP      DEFINE 1           =1 IF INSTRUCTION IS A JP ON *MRC*    SQZ          42     A
SQZ                DEFINE 5                                                 SQZ          43     A
SQZ        RS      DEFINE 18          *RS* DEFINITION ADDRESS               SQZ          44     A
SQZ        STC     DEFINE 18          STORE CHAIN ( POINTER TO PRECEEDING STORES  SQZ  45     A
SQZ        IC      DEFINE 18          MEMORY REF / MISC INSTRUCTION CHAINS  SQZ          46     A
SQZ                                                                         CCG17        1     A
SQZ        CA      DEQU   RS          SAVED CA OF *LDC* INSTRUCTIONS        CCG17        2          I
          -CCG18
SQZ        CA      DEQU   STC         SAVED CA OF LDC INSTRUCTION           CCG18       146     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                SPACE  2                                                    SQZ          47    A
SQZ       SQZ      TITLE  TABLE FORMATS, FLAGS                                 SQZ          48    A
SQZ      **        RN. - R-NUMBER DEFINITION/EQUIVALENCE TABLE FORMAT          SQZ          49    A
SQZ                                                                            SQZ          50    A
SQZ                DESCRIBE RN.,60                                             SQZ          51    A
SQZ       Z        DEFINE 1           RI DEFINES A ZERO                        SQZ          52    A
SQZ       C        DEFINE 1           SMALL CONSTANT ( S / FMA )               SQZ          53          I
         -CCG17
SQZ       C        DEFINE 1           CONSTANT ( S, FMA, CLR, LDC, ETC. )      CCG17         3    A
SQZ       SA       DEFINE 1           RI IS RESULT OF SA  RI,RJ                SQZ          54    A
SQZ       IAC      DEFINE 1           IA/IS HAS A CONSTANT OPERAND             SQZ          55    A
SQZ       INC      DEFINE 1           IA/IS IS AN INCREMENT ( I = I+CON )      SQZ          56    A
SQZ       XMT      DEFINE 1           RI IS THE RESULT OF AN XMT               SQZ          57    A
SQZ       XMTC     DEFINE 1           RI IS THE RESULT OF AN XMTC              SQZ          58    A
SQZ       PZ       DEFINE 1           PACKED ZERO                              SQZ          59    A
SQZ       NR       DEFINE 1           NORMALIZE NOT REQUIRED                   SQZ          60    A
SQZ       FMA      DEFINE 1           "0 IF FMA CHANGED TO A SET               SQZ          61    A
SQZ                DEFINE 4                                                    SQZ          62    A
SQZ       USES     DEFINE 10          USES COUNT                               SQZ          63    A
SQZ       TP       DEFINE 18          *TXT* ADDR OF INSTRUCTION DEFINING THE RI SQZ         64    A
SQZ       RN       DEFINE 18          R-NUMBER / EQUIVALENT                    SQZ          65    A
SQZ                                                                            SQZ          66    A
SQZ       CZ       DEQU   C,2         CONSTANT AND ZERO BITS                   SQZ          67    A
SQZ       CIA      DEQU   IAC,3       C, SA , IAC BITS                         SQZ          68    A
SQZ       IACI     DEQU   INC,2       IAC & INC                                SQZ          69    A
SQZ       FLAGS    SPACE  3                                                    SQZ          70    A
SQZ      *         FLAGS                                                       SQZ          71    A
SQZ                                                                            SQZ          72    A
SQZ       BIAS     BSSZ   1           *RNI* BIAS , =0 DURING PHASE 1 OF OPT=2  SQZ          73    A
SQZ                                                                            SQZ          74    A
SQZ                USE    /TABLES/                                            SQZ          75          I
         -CCG1
SQZ                USE    /CCG.SCR/                                           CCG1        261          I
         -CCG18
SQZ                USE    /CCGSCR/                                            CCG18       147    A
SQZ       RNBT     BSS    2           R-NUMBER BIAS TABLE                     SQZ          76    A
SQZ                                                                            SQZ          77    A
SQZ       MRC      BSS    1           MEMORY REFERENCE CHAIN POINTER          SQZ          78    A
SQZ       MIC      BSS    1           MISCELLANEOUS INSTRUCTION CHAIN         SQZ          79    A
SQZ       CRJC     BSS    1           CONDITIONAL  RJ CHAIN                   CCG3          1    A
SQZ       SXTC     BSS    1           SXT CHAIN                               SQZ          80    A
SQZ       FAC      BSS    1           FUNCTION ARG CHAIN                      CCGA083       6    A
SQZ       RJ3C     BSS    1           RJ3 CHAIN                               CCGA083       7    A
SQZ       LDCC     BSS    1           LDC CHAIN                               CCG17         4    A
SQZ       N.CHAIN  EQU    *-MRC                                               SQZ          81    A
SQZ       EOQA     BSS    1           ADDR OF FINAL *EOQ*                     SQZ          82          I
         -CCG17
SQZ       IXF      BSS    1           "0 IF *LDC* & PS OR *S* & SO ^= 0       CCG17         5    A
SQZ       LDCN     BSS    1           LDC COUNT                               CCG17         6    A
SQZ       RSS      BSS    1           RS SEPARATION FLAG                      SQZ          83    A
SQZ       LSTA     BSS    1           LAST *ST* ADDRESS ( LINK WORD )         SQZ          84          I
         -FCC0957

         0        1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ       STC      BSS    1           ADDRESS OF FINAL STORE ( ^L2 )                SQZ          85     A
SQZ       JPXA     BSS    1           R1 ADDRESS OF LAST *JPX*                      SQZ          86     A
SQZ       RJ3T     BSS    1           TEMP                                          CCGA083       8     A
SQZ       FACS     BSS    1           TEMP                                          CCGA083       9     A
SQZ       CAS      BSS    1           CA'S OF STORES                                CCGA083      10     A
SQZ       M.STK    EQU    40                                                        CCGA083      11     A
SQZ       STK      BSS    M.STK       ADDRESS STACK FOR RJ3 SQZ                     CCGA083      12     A
SQZ                EXT    N.STMAX                                                   CCGA083      13     A
SQZ       STBASE   EQU    N.STMAX                                                   CCGA083      14     A
SQZ                USE    0                                                         SQZ          87     A
SQZ                SPACE  2                                                         SQZ          88     A
SQZ      *         REGISTER ASSIGNMENTS                                             SQZ          89     A
SQZ                                                                                 SQZ          90     A
SQZ       TXT      MICRO  1,,/A0/     REGISTER HOLDING FWA OF BLOCK                 SQZ          91     A
SQZ       OC       MICRO  1,,/B3/     B REGISTER HOLDING OPCODE                     SQZ          92     A
SQZ       RN       MICRO  1,,/B7/     B REGISTER HOLDING BASE OF *RND*              SQZ          93     A
SQZ      *         (B1) = 1                                                         SQZ          94     A
SQZ      *         (B2) = 2                                                         SQZ          95     A
SQZ      *         (X0) = MASK FOR RI FIELD                                         SQZ          96     A
SQZ      *         THROUGHOUT THE MASTER SQUEEZE LOOP                               SQZ          97     A
SQZ      *         (A4,X4) = DESCRIPTOR OF CURRENT RLIST INSTRUCTION                SQZ          98     A
SQZ      *         (A5) = ADDRESS OF INSTRUCTION BEING INSPECTED                    SQZ          99     A
SQZ      *         (B6) = RI OF CURRENT INSTRUCTION                                 SQZ         100     A
SQZ       SQZB     TITLE  SIE / SQZB - OPT=2 ENTRY POINTS TO SQUEEZE                SQZ         101     A
SQZ      **        SIE - SQUEEZE INCREMENT EXPRESSION                               SQZ         102     A
SQZ      *                                                                          SQZ         103     A
SQZ      *         EXIT   INSTRUCTIONS IN *TXT* RENUMBERED SO THAT THEY WILL        SQZ         104     A
SQZ      *                BE IN CANONICAL ORDER WHEN THEY ARE ADDED TO THE          SQZ         105     A
SQZ      *                LOOP HOLDING BLOCK.                                       SQZ         106     A
SQZ                                                                                 SQZ         107     A
SQZ       SIE      ROUTINE                                                          SQZ         108     A
SQZ                SA5    SIE                                                       SQZ         109     A
SQZ                SA2    =XL.MOD                                                   SQZ         110     A
SQZ                SA3    =XL.HB                                                    SQZ         111     A
SQZ                MX6    0           TSS = 0                                       SQZ         112     A
SQZ                IX4    X2+X3                                                     SQZ         113     A
SQZ                SX7    X4-8        BIAS = L.HB + L.MOD-8                         SQZ         114     A
SQZ                EQ     SQZB1                                                     SQZ         115     A
SQZ       SQZB     SPACE  3                                                         SQZ         116     A
SQZ      **        SQZB - OPT=2 ENTRY TO SQUEEZE                                    SQZ         117     A
SQZ      *                                                                          SQZ         118     A
SQZ      *                (X6) = TSS , =1 IF *TST* SQUEEZING IS TO BE DONE.         SQZ         119     A
SQZ                                                                                 SQZ         120     A
SQZ       SQZB     ROUTINE                                                          SQZ         121     A
SQZ                SA5    SQZB                                                      SQZ         122     A
SQZ                SX7    0           BIAS = 0                                      SQZ         123     A
SQZ                SA7    STLDS       STLDS = 0                                     SQZ         124     A
SQZ                                                                                 SQZ         125     A
SQZ       SQZB1    SA1    L.TXT                                                     SQZ         126     A
SQZ                SA2    SQZBA+X6                                                  SQZ         127     A
SQZ                SA7    BIAS                                                      SQZ         128     A
SQZ                BX6    X5                                                        SQZ         129     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                LX7    X2                                                    SQZ          130     A
SQZ                SA6    SBB         [SBB] = EXIT ADDRESS                      SQZ          131     A
SQZ                SA7    SBB.JT      [SBB.JT] = *EOQ* PROCESSOR ADDRESS        SQZ          132     A
SQZ                                                                             SQZ          133           I
          -FCC0509
SQZ                ALLOC  RND,X1-6    ALLOC( RND , L.TXT-6 )                    SQZ          134     A
SQZ                SA5    O.TXT                                                 SQZ          135     A
SQZ                MX0    -R1.RIL                                               SQZ          136     A
SQZ                S"RN"  X2                                                    SQZ          137     A
SQZ                SB2    B1+B1                                                 SQZ          138     A
SQZ                S"TXT" X5                                                    SQZ          139     A
SQZ                EQ     SBB5A                                                 SQZ          140     A
SQZ                                                                             SQZ          141     A
SQZ       SQZBA    BSS    0                                                     SQZ          142     A
SQZ                LOC    0                                                     SQZ          143     A
SQZ                EQ     .EOQ        NO *TST* SQZ                              SQZ          144     A
SQZ       -        VFD    12/1S10+(EOQ,18/0                                     CCG22          1     A
SQZ                EQ     STS                                                   SQZ          145     A
SQZ       -        VFD    12/1S10+(EOQ,18/0                                     CCG22          2     A
SQZ                LOC    *O                                                    SQZ          146     A
SQZ                                                                             SQZ          147           I
          -CCGNC
SQZ                QUAL                                                         SQZ          148           I
          -CCGNC
SQZ                                                                             SQZ          149           I
          -CCGNC
SQZ       STLDS    ENTRY. 0           ST/LD SQUEEZE FLAG, "0 IF ST/LD SQZ DONE SQZ          150           I
          -CCGNC
SQZ       SUVA     ENTRY. 0           SQUEEZED USE *BV* ADDR ( *AUV* CALL )     FCC0509       46           I
          -CCGNC
SQZ                                                                             SQZ          151           I
          -CCGNC
SQZ                ENTRY  SIE,SQZB                                              SQZ          152           I
          -CCGNC
SQZ       SIE      EQU    /SQUEEZE/SIE                                          SQZ          153           I
          -CCGNC
SQZ       SQZB     EQU    /SQUEEZE/SQZB                                         SQZ          154           I
          -CCGNC
SQZ                                                                             SQZ          155           I
          -CCGNC
SQZ                QUAL   SQUEEZE                                               SQZ          156           I
          -CCGNC
SQZ       STS      TITLE  STS - SQUEEZE TEMP STORES TO ELIMINATE REDUNDANT ONES SQZ          157     A
SQZ       **       STS - SQUEEZE TEMP STORES ( *TST*"S ) TO ELIMINATE REDUNDANT SQZ          158     A
SQZ       *        ONE"S AND SET THE *CA* FIELD OF THE NON-REDUNDANT ONES.      SQZ          159     A
SQZ       *                                                                     SQZ          160     A
SQZ       *        ENTRY  FROM *SBB.JT* WHEN *EOQ* IS ENCOUNTERED.              SQZ          161     A
SQZ       *        EXIT   TO *EOQ* PROCESSOR WHEN FINISHED                      SQZ          162     A
SQZ                                                                             SQZ          163     A
SQZ       STS      SA1    =XN.GT      N = N.GT      */ NUMBER OF ASSIGNED IT. S SQZ          164     A
SQZ                SA5    STC                                                   SQZ          165     A
SQZ                SB2    OC.TST                                                SQZ          166     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                SB4    B0          USC = 0       */ UNASSIGNED *TST* CHAIN          SQZ        167    A
SQZ                SB5    X5          I = STC                                          SQZ        168    A
SQZ                SB6    "TXT"+3     TB = O.TXT + 3                                   SQZ        169    A
SQZ                SA0    B0          ASC = 0       */ ASSIGNED *TST* CHAIN            SQZ        170    A
SQZ                MX7    1                                                            SQZ        171    A
SQZ                EQ     STS2                                                         SQZ        172    A
SQZ                                                                                   SQZ        173    A
SQZ       *        SCAN *STC* AND FORM CHAINS OF ASSIGNED AND UNASSIGNED *TST*"S       SQZ        174    A
SQZ                                                                                   SQZ        175    A
SQZ        STS1    SX6    A0                                                           SQZ        176    A
SQZ                SA6    A5          [LI] = ASC    */ LINK LAST TO THIS               SQZ        177    A
SQZ                SA0    A5+         ASC = LI      */ CHAIN POINTS TO THIS            SQZ        178    A
SQZ                                                                                   SQZ        179    A
SQZ        STS2    SA5    B5          LI = I                                           SQZ        180    A
SQZ                ZR     B5,STS3     IF I = 0      */ END OF CHAIN                    SQZ        181    A
SQZ                LX5    -I.STCP                                                      SQZ        182    A
SQZ                SA3    A5-3        R1 = LI - 3                                      SQZ        183    A
SQZ                SA4    A3+B1       R2 = R1 + 1                                      SQZ        184    A
SQZ                SB5    X5          I = STC[LI]                                      SQZ        185    A
SQZ                UX6    B3,X3                                                        SQZ        186    A
SQZ                NE     B3,B2,STS2  IF OC[R1] " OC.TST                               SQZ        187    A
SQZ                                                                                   SQZ        188    A
SQZ                LX4    -IH.CAP                                                      SQZ        189    A
SQZ                SX2    X4          C = CA[R2]                                       SQZ        190    A
SQZ                IX6    X2-X1                                                        SQZ        191    A
SQZ                BX5    -X0*X3                                                       SQZ        192    A
SQZ                SA7    B6+X5       LIP = TB + RI[R1];  [LIP] = 1S59  */ NOT AS      SQZ        193    A
SQZ                MI     X6,STS1     IF C < N      */ TEMP NUMBER ASSIGNED            SQZ        194    A
SQZ                                                                                   SQZ        195    A
SQZ                SX6    B4                                                           SQZ        196    A
SQZ                SA6    A5          [LI] = USC                                       SQZ        197    A
SQZ                SB4    A5          USC = LI                                         SQZ        198    A
SQZ                EQ     STS2                                                         SQZ        199    A
SQZ                                                                                   SQZ        200    A
SQZ        STS3    SA2    =XO.TET                                                      SQZ        201    A
SQZ                SB3    A0          I = ASC                                          SQZ        202    A
SQZ                SB5    X2                        (B5) = [O.TET]                     SQZ        203    A
SQZ                ZR     B3,STS7     IF I = 0      */ NO ASSIGNED TET"S               SQZ        204    A
SQZ                                                                                   SQZ        205    A
SQZ       *        SET THE *CA*"S OF THE ASSIGNED *TST*"S IN THE LINK WORD OF          SQZ        206    A
SQZ       *        THE INSTRUCTION DEFINING THE *RI* IN THE *TST*.                     SQZ        207    A
SQZ                                                                                   SQZ        208    A
SQZ        STS4    SA3    B3-3        R1 = I - 3                                       SQZ        209    A
SQZ                SA4    A3+B1       R2 = R1 + 1                                      SQZ        210    A
SQZ                BX6    -X0*X3      R = RI[R1]                                       SQZ        211    A
SQZ                SA2    B6+X6       LIP = TB + R;  K = [LIP]                         SQZ        212    A
SQZ                LX4    -IH.CAP                                                      SQZ        213    A
SQZ                SX7    X4          C = CA[R2]                                       SQZ        214    A
SQZ                MI     X2,STS5     IF K < 0      */ NOT EQUIVALENCED                SQZ        215    A
SQZ                                                                                   SQZ        216    A
SQZ       *        EQUIVALENCE THE CELLS IF THEY ARE BOTH INVARIANT TEMPS              SQZ        217    A
SQZ                                                                                   SQZ        218    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ              SA3    B5+X7                                                  SQZ          219     A
SQZ              SX5    B1                                                     CCGA049       16     A
SQZ              LX5    T.CALP                                                 CCGA049       17     A
SQZ              BX4    X3*X5                                                  CCGA049       18     A
SQZ              NZ     X4,STS5     IF CA FIELD STILL IN USE                   CCGA049       19     A
SQZ              SA5    B5+X2                                                  SQZ          220     A
SQZ              BX4    X3*X5                                                  SQZ          221     A
SQZ              LX4    59-T.INVP                                              SQZ          222     A
SQZ              PL     X4,STS5     IF ^( INV[TET(C)] & INV[TET(K)] )          SQZ          223     A
SQZ                                                                            SQZ          224     A
SQZ              BX7    X0*X3                                                  SQZ          225     A
SQZ              SX5    B1                                                     SQZ          226     A
SQZ              BX7    X7+X2       CA[TET(C)] = K                             SQZ          227     A
SQZ              LX5    T.EQVP                                                 SQZ          228     A
SQZ              BX7    X5+X7       EQV[TET(C)] = 1                            SQZ          229     A
SQZ              SA7    A3                                                     SQZ          230     A
SQZ              SA2    "RN"+X6     RNI = RND(R)                               SQZ          231     A
SQZ              LX5    RN.USESP-T.EQVP                                        SQZ          232     A
SQZ              IX6    X2-X5       USES[RNI] = USES[RNI] - 1                  SQZ          233     A
SQZ              SA6    A2                                                     SQZ          234     A
SQZ              MX6    0                                                      SQZ          235     A
SQZ              SA6    A4+B1       DI = R2 + 1;  [DI] = 0  */ KILL ST         SQZ          236     A
SQZ              EQ     STS6                                                   SQZ          237     A
SQZ                                                                            SQZ          238     A
SQZ       STS5   SA7    A2+         [LIP] = C     */ SET ASSIGNMENT            SQZ          239     A
SQZ                                                                            SQZ          240     A
SQZ       STS6   SA5    B3          LI = [I]                                   SQZ          241     A
SQZ              SB3    X5          I = [LI]                                   SQZ          242     A
SQZ              NZ     B3,STS4     IF I " 0                                   SQZ          243     A
SQZ                                                                            SQZ          244     A
SQZ      *       ASSIGN TEMP NUMBERS TO THE UNASSIGNED *TST*"S                 SQZ          245     A
SQZ                                                                            SQZ          246     A
SQZ       STS7   SX2    B1          I = USC                                    SQZ          247     A
SQZ              LX2    RN.USESP                                               SQZ          248     A
SQZ              ZR     B4,STS0     IF I = 0      */ NO UNASSIGNED TEMPS       SQZ          249     A
SQZ                                                                            SQZ          250     A
SQZ       STS8   SA3    B4-3        R1 = I - 3                                 SQZ          251     A
SQZ              SA4    A3+B1       R2 = R1 + 1                                SQZ          252     A
SQZ              BX6    -X0*X3                                                 SQZ          253     A
SQZ              SA5    B6+X6       LIP = TB + RI[R1];  K = [LIP]              SQZ          254     A
SQZ              LX4    -IH.CAP     C = CA[R2]                                 SQZ          255     A
SQZ              MI     X5,STS9     IF K < 0      */ NO TEMP NUMBER ASSIGNED   SQZ          256     A
SQZ                                                                            SQZ          257     A
SQZ      *       *TST* IS REDUNDANT, ELIMINATE IT                              SQZ          258     A
SQZ                                                                            SQZ          259     A
SQZ              SA3    "RN"+X6     R = RND(RI[R1])                            SQZ          260     A
SQZ              MX7    0                                                      SQZ          261     A
SQZ              IX6    X3-X2       USES[R] = USES[R] - 1                      SQZ          262     A
SQZ              SA7    A4+B1       DI = R2 + 1;  [DI] = 0                     SQZ          263     A
SQZ              SA6    A3                                                     SQZ          264     A
SQZ              EQ     STS10                                                  SQZ          265     A
SQZ                                                                            SQZ          266     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ       *          *TST* NOT REDUNDANT, ASSIGN IT A TEMP NUMBER                    SQZ          267     A
SQZ                                                                                  SQZ          268     A
SQZ        STS9      BX7    X1                                                       SQZ          269     A
SQZ                  BX5    X0*X4                                                    SQZ          270     A
SQZ                  SA7    A5          [LIP] = N     ASSIGN TEMP NUMBER             SQZ          271     A
SQZ                  IX6    X5+X1       CA[R2] = N                                   SQZ          272     A
SQZ                  LX6    IH.CAP                                                   SQZ          273     A
SQZ                  BX5    X1          K = N                                        SQZ          274     A
SQZ                  SA6    A4                                                       SQZ          275     A
SQZ                  SX1    X1+B1       N = N + 1                                    SQZ          276     A
SQZ                                                                                  SQZ          277     A
SQZ        STS10     SA3    B5+X4                                                    SQZ          278     A
SQZ                  BX6    X3+X5       CA[TET(C)] = K  */ SET FINAL *CA* IN *TET*   SQZ          279     A
SQZ                  SX3    1                                                        CCGA049       20     A
SQZ                  LX3    T.CALP                                                   CCGA049       21     A
SQZ                  BX6    X6+X3       CAL[TET(C)]=1  */ LOCK CA FIELD              CCGA049       22     A
SQZ                  SA6    A3                                                       SQZ          280     A
SQZ                  BX7    X5-X4                                                    SQZ          281     A
SQZ                  SB3    X7                                                       SQZ          282     A
SQZ                  ZR     B3,STS11    IF C = K                                     SQZ          283     A
SQZ                                                                                  SQZ          284     A
SQZ                  SA5    B5+X5                                                    SQZ          285     A
SQZ                  MX7    1                                                        SQZ          286     A
SQZ                  LX7    1+T.INVP                                                 SQZ          287     A
SQZ                  BX6    X7*X6                                                    SQZ          288     A
SQZ                  BX5    -X7*X5                                                   SQZ          289     A
SQZ                  BX6    X6+X5       INV[TET(K)] = INV[TET(C)]                    SQZ          290     A
SQZ                  SA6    A5                                                       SQZ          291     A
SQZ                                                                                  SQZ          292     A
SQZ        STS11     SA4    B4          LI = I                                       SQZ          293     A
SQZ                  SB4    X4          I = [LI]      */ NEXT ON CHAIN               SQZ          294     A
SQZ                  NZ     B4,STS8     IF I " 0                                     SQZ          295     A
SQZ                                                                                  SQZ          296     A
SQZ        STS0      BX7    X1          N.GT = N      */ NEXT AVAIL TEMP NO          SQZ          297     A
SQZ                  SA2    SQZBA                                                    SQZ          298     A
SQZ                  SB2    B1+B1                                                    SQZ          299     A
SQZ                  SA7    N.GT                                                     SQZ          300     A
SQZ                  BX6    X2          [SBB.JT] = [SQZBA]  */ RESTORE *EOQ* PROC    SQZ          301     A
SQZ                  S"TXT" B6-3                                                     SQZ          302     A
SQZ                  SA6    SBB.JT                                                   SQZ          303     A
SQZ                  EQ     .EOQ                                                     SQZ          304     A
SQZ        SQZ       TITLE  INITIALIZATION                                           SQZ          305     A
SQZ       **         SBB - SQUEEZE EXTENDED BASIC BLOCK. ENTRY TO SQZ FROM PROSEQ.   SQZ          306           I
          -CCG1
SQZ       **         SBB - SQUEEZE EXTENDED BASIC BLOCK. ENTRY TO SQZ FROM CG.PAS.   CCG1          262           I
          -CCGNC
SQZ       **         SBB - SQUEEZE EXTENDED BASIC BLOCK. ENTRY TO SQZ FROM CG$PAS.   CCGNC         751     A
SQZ       *                                                                          SQZ          307     A
SQZ       *          ENTRY  (BRN) = BASE,LIMIT R-NUMBERS IN SEQUENCE                SQZ          308           I
          -CCG1
SQZ       *                 (BIR) = BASE LIMIT INTERMEDIATE R-NUMBERS IN SEQUENCE. SQZ          309           I
          -CCG1

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ      *         ENTRY   (CG.BRN) = BASE,LIMIT R-NUMBERS IN SEQUENCE                CCG1      263        I
         -CCGNC
SQZ      *         ENTRY   (CC$BRN) = BASE,LIMIT R-NUMBERS IN SEQUENCE                CCGNC     752    A
SQZ      *                 (CG.BIR) = BASE,LIMIT INTERMEDIATE R-NUMBERS               CCG1      264        I
         -CCGNC
SQZ      *                 (CC$BIR) = BASE,LIMIT INTERMEDIATE R-NUMBERS               CCGNC     753    A
SQZ       SBB      ROUTINE                                                            SQZ       310    A
SQZ                SA1     BRN                                                        SQZ       311        I
         -CCG1
SQZ                SA3     BIR                                                        SQZ       312        I
         -CCG1
SQZ                IFEQ    HC.ROL,0                                                   CCG1      265        I
         -CCG9
SQZ       .ROL     IFEQ    HC.ROL,0                                                   CCG9      126    A
SQZ                SA1     =XCG.BRN+1                                                 CCG1      266        I
         -CCGNC
SQZ                SA1     =XCC$BRN+1                                                 CCGNC     754    A
SQZ                AX1     2                                                          CCG22       3    A
SQZ       .ROL     ELSE                                                               CCG1      267    A
SQZ                                                                                   CCG1      268    A
SQZ                SA1     =XCG.BRN                                                   CCG1      269        I
         -CCGNC
SQZ                SA1     =XCC$BRN                                                   CCGNC     755    A
SQZ                SA3     =XCG.BIR                                                   CCG1      270        I
         -CCGNC
SQZ                SA3     =XCC$BIR                                                   CCGNC     756    A
SQZ                SA2     A1+B1                                                      SQZ       313    A
SQZ                SA4     A3+B1                                                      SQZ       314    A
SQZ                IX2     X2-X1       DEL(R) = BRN(2) - BRN(1)                       SQZ       315        I
         -CCG1
SQZ                IX2     X2-X1       DEL(R) = CG.BRN(2) - CG.BRN(1)                 CCG1      271        I
         -CCGNC
SQZ                IX2     X2-X1       DEL(R) = CC$BRN(2) - CC$BRN(1)                 CCGNC     757    A
SQZ                SX0     B1+B1                                                      SQZ       316    A
SQZ                IX4     X4-X3       DEL(IR) = BIR(2) - BIR(1)                      SQZ       317        I
         -CCG1
SQZ                IX4     X4-X3       DEL(IR) = CG.BIR(2) - CG.BIR(1)                CCG1      272        I
         -CCGNC
SQZ                IX4     X4-X3       DEL(IR) = CC$BIR(2) - CC$BIR(1)                CCGNC     758    A
SQZ                IX7     X2+X4                                                      SQZ       318    A
SQZ                IX6     X0-X1                                                      SQZ       319    A
SQZ                IX1     X7+X0                                                      SQZ       320    A
SQZ                IX7     X0-X3                                                      SQZ       321    A
SQZ                SA6     RNBT        RNBT(0) = 2 - BRN(1)                           SQZ       322        I
         -CCG1
SQZ                SA6     RNBT        RNBT(0) = 2 - CG.BRN(1)                        CCG1      273        I
         -CCGNC
SQZ                SA6     RNBT        RNBT(0) = 2 - CC$BRN(1)                        CCGNC     759    A
SQZ                IX7     X2+X7                                                      SQZ       323    A
SQZ                SA7     A6+B1       RNBT(1) = 2 + DEL(R) - BIR(1)                  SQZ       324        I
         -CCG1
SQZ                SA7     A6+B1       RNBT(1) = 2 + DEL(R) - CG.BIR(1)               CCG1      274        I

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

          -CCGNC
SQZ                 SA7    A6+B1       RNBT(1) = 2 + DEL(R) - CC$BIR(1)          CCGNC      760    A
SQZ        .ROL     ENDIF                                                        CCG1       275    A
SQZ                                                                             CCG1       276    A
SQZ                 ALLOC  RND,X1      ALLOC( RND , DEL(IR)+DEL(R)+2 )           SQZ        325    A
SQZ                                                                             SQZ        326    A
SQZ       *         NOTE - ABSOLUTE ADDRESSES ARE USED THROUGHOUT THE REST OF   SQZ        327    A
SQZ       *         *SBB*, HENCE NO TABLE ALLOCATION REQUESTS MAY BE MADE.      SQZ        328    A
SQZ                                                                             SQZ        329    A
SQZ                 SA1    O.TXT                                                SQZ        330    A
SQZ                 S"RN"  X2          (B7) = O.RND                             SQZ        331    A
SQZ                 S"TXT" X1          (A0) = O.TXT                             SQZ        332    A
SQZ                 SB2    B1+B1             (B2) = 2                            CCG18      148    A
SQZ                 MX0    -R1.RIL           (X0) = R-NUMBER EXTRACTOR MASK      CCG18      149         I
          -CCG22
SQZ        SBB      SPACE  3           SBB                                      SQZ        333    A
SQZ        .ROL     IFNE   HC.ROL,0                                             CCG1       277    A
SQZ       **        REPLACE ALL OCCURRENCES OF R-NUMBERS BY INDICES TO *RND*.   SQZ        334    A
SQZ       *         MAP R,R+1,...,IR,IR+1,... INTO  2,3,4,...                   SQZ        335    A
SQZ       *         NOTE - LOOP IS STRUCTURED TO BE *INSTACK* FOR TYPE I PROCESSIN SQZ      336    A
SQZ                                                                             SQZ        337    A
SQZ                 MX0    -R1.RIL           (X0) = R-NUMBER EXTRACTOR MASK      CCG22        4    A
SQZ                 SB5    R1.RIL-1                                             SQZ        338    A
SQZ                 SB6    RNBT                                                 SQZ        339    A
SQZ                 SB2    B1+B1                                                SQZ        340         I
          -CCG18
SQZ                 MX0    -R1.RIL     (X0) = MASK FOR EXTRACTING R-NUMBERS     SQZ        341         I
          -CCG18
SQZ                 MX1    1                                                    SQZ        342    A
SQZ                 SA4    "TXT"+B2    DI = O.TXT + 2                           SQZ        343    A
SQZ                 EQ     SBB2                                                 SQZ        344    A
SQZ                                                                             SQZ        345    A
SQZ       *         TYPE II/III                                                 SQZ        346    A
SQZ                                                                             SQZ        347    A
SQZ        SBB1     BX3    -X0*X5      RI = RI[R1]                              SQZ        348    A
SQZ                 AX7    B5,X3                                                SQZ        349    A
SQZ                 SA2    X7+B6       BIAS = RNBT( SHIFT(RI,-15) )             SQZ        350    A
SQZ                 SA3    A5+B1       R2 = R1 + 1                              SQZ        351    A
SQZ                 IX6    X2+X5       RI = RI + BIAS                           SQZ        352    A
SQZ                 SA6    A5                                                   SQZ        353    A
SQZ                 LX3    -IH.RFP                                              SQZ        354    A
SQZ                 BX6    -X0*X3                                               SQZ        355    A
SQZ                 AX7    B5,X6                                                SQZ        356    A
SQZ                 SB4    X6                                                   SQZ        357    A
SQZ                 LE     B4,B1,SBB2  IF RF @ 1                                SQZ        358    A
SQZ                 SA2    X7+B6                                                SQZ        359    A
SQZ                 IX6    X2+X3       RF = RF + BIAS                           SQZ        360    A
SQZ                 LX6    IH.RFP                                               SQZ        361    A
SQZ                 SA6    A3                                                   SQZ        362    A
SQZ                                                                             SQZ        363    A
SQZ        SBB2     SA5    A4+B2       R1 = DI + 2                              SQZ        364    A
SQZ                 SA4    A5+B2       DI = R1 + 2                              SQZ        365    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                LX4    58-D.TYP                                                  SQZ        366    A
SQZ                IX6    X4+X4                                                     SQZ        367    A
SQZ                BX7    X4-X6                                                     SQZ        368    A
SQZ                MI     X7,SBB1     IF TYPE[DI] = II ! III                        SQZ        369    A
SQZ                MI     X4,SBB5     IF TYPE[DI] = IV                              SQZ        370    A
SQZ                                                                                SQZ        371    A
SQZ      *         TYPE I INSTRUCTION                                              SQZ        372    A
SQZ                                                                                SQZ        373    A
SQZ                UX7    "OC",X5                                                   SQZ        374    A
SQZ                SX6    "OC"+6000B  RWD = OC[R1] + 6000B  */ PACKED OC + TERM     SQZ        375    A
SQZ                LX5    -R1.RJP     RNUMS = RJ/RK/RI                              SQZ        376    A
SQZ                                                                                SQZ        377    A
SQZ       SBB3     BX3    -X0*X5                                                    SQZ        378    A
SQZ                SB4    X3                                                        SQZ        379    A
SQZ                AX4    B5,X3                                                     SQZ        380    A
SQZ                SA2    X4+B6       BIAS = RNBT( SHIFT(RN,-15) )                  SQZ        381    A
SQZ                LX6    R1.RIL      RWD = SHIFT(RWD,16)                           SQZ        382    A
SQZ                LE     B4,B1,SBB4  IF RN @ 1                                     SQZ        383    A
SQZ                IX3    X2+X3       RN = RN + BIAS                                SQZ        384    A
SQZ       SBB4     LX5    R1.RIL      RNUMS = SHIFT(RNUMS,16)                       SQZ        385    A
SQZ                BX6    X6+X3                                                     SQZ        386    A
SQZ                PL     X6,SBB3     IF RWD > 0                                    SQZ        387    A
SQZ                BX7    -X1*X6      [R1] = TYI(OC,RJ,RK,RI)                       SQZ        388    A
SQZ                SA7    A5                                                        SQZ        389    A
SQZ                EQ     SBB2                                                      SQZ        390    A
SQZ                                                                                SQZ        391    A
SQZ       SBB5     UX7    "OC",X5                                                   SQZ        392    A
SQZ                NZ     "OC",SBB2   IF OC[R1] " EOQ                               SQZ        393    A
SQZ       .ROL     ELSE                                                              CCG22        5    A
SQZ                                                                                CCG22        6    A
SQZ      *         REPLACE ALL OCCURANCES OF R-NUMBERS BY INDICES TO *RND*         CCG22        7    A
SQZ      *         MAP 0,1,4,8,....    INTO  0,1,2,3,...                           CCG22        8    A
SQZ                                                                                CCG22        9    A
SQZ                MX0    -R1.RIL+1                                                 CCG22       10         I
          -CCG28
SQZ                LX0    1                                                         CCG22       11         I
          -CCG28
SQZ                MX0    -R1.RIL                                                   CCG28       37    A
SQZ                MX1    1                                                         CCG22       12    A
SQZ                SA4    "TXT"+B2    DI = O.TXT + 2                                CCG22       13    A
SQZ                EQ     SBB2                                                      CCG22       14    A
SQZ                                                                                CCG22       15    A
SQZ      *         TYPE II/III                                                     CCG22       16    A
SQZ                                                                                CCG22       17    A
SQZ       SBB1     BX3    -X0*X5                                                    CCG22       18    A
SQZ                ZR     X3,SBB1A    IF R = 0                                      CCG22       19    A
SQZ                AX3    2                                                         CCG22       20    A
SQZ                BX5    X0*X5                                                     CCG22       21    A
SQZ                SX3    X3+B1       R = R/4 + 1                                   CCG22       22    A
SQZ                IX6    X3+X5                                                     CCG22       23    A
SQZ                SA6    A5                                                        CCG22       24    A
SQZ       SBB1A    SA2    A5+B1       R2 = R1 + 1                                   CCG22       25    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                LX2    -IH.RFP                                          CCG22        26     A
SQZ                BX3    -X0*X2                                           CCG22        27     A
SQZ                ZR     X3,SBB2     IF R @ 1                             CCG22        28          I
          -CCG28
SQZ                AX3    2                                                CCG22        29          I
          -CCG28
SQZ                AX3    2                                                CCG28        38     A
SQZ                ZR     X3,SBB2     IF R < 4                             CCG28        39     A
SQZ                BX2    X0*X2                                            CCG22        30     A
SQZ                SX3    X3+B1       R = R/4 + 1                          CCG22        31     A
SQZ                IX6    X3+X2                                            CCG22        32     A
SQZ                LX6    IH.RFP                                           CCG22        33     A
SQZ                SA6    A2                                               CCG22        34     A
SQZ                                                                       CCG22        35     A
SQZ       SBB2     SA5    A4+B2       R1 = DI + 2                          CCG22        36     A
SQZ                SA4    A5+B2       DI = R1 + 2                          CCG22        37     A
SQZ                LX4    58-D.TYP                                         CCG22        38     A
SQZ                IX6    X4+X4                                            CCG22        39     A
SQZ                BX7    X4-X6                                            CCG22        40     A
SQZ                MI     X7,SBB1     IF TYPE[DI] = II ! III               CCG22        41     A
SQZ                MI     X4,SBB5     IF TYPE[DI] = IV                     CCG22        42     A
SQZ                                                                       CCG22        43     A
SQZ       *        TYPE I INSTRUCTION                                     CCG22        44     A
SQZ                                                                       CCG22        45     A
SQZ                UX7    "OC",X5                                          CCG22        46     A
SQZ                SX6    "OC"+6000B  RWD = OC[R1] + 6000B  */ PACKED OC + TERM  CCG22  47     A
SQZ                LX5    -R1.RJP     RNUMS = RJ/RK/RI                     CCG22        48     A
SQZ                                                                       CCG22        49     A
SQZ       SBB3     BX3    -X0*X5                                           CCG22        50     A
SQZ                LX6    R1.RIL                                           CCG22        51     A
SQZ                ZR     X3,SBB4     IF R = 0                             CCG22        52          I
          -CCG28
SQZ                AX4    B1,X3                                            CCG28        40     A
SQZ                ZR     X4,SBB4     IF R < 2                             CCG28        41     A
SQZ                AX3    2                                                CCG22        53     A
SQZ                SX3    X3+1        R = R/4 + 1                          CCG22        54          I
          -CCG28
SQZ                SX3    X3+B1       R = R/4 + 1                          CCG28        42     A
SQZ       SBB4     LX5    R1.RIL      RNUMS = SHIFT(RNUMS,16)              CCG22        55     A
SQZ                BX6    X6+X3                                            CCG22        56     A
SQZ                PL     X6,SBB3     IF RWD > 0                           CCG22        57     A
SQZ                BX7    -X1*X6      [R1] = TYI(OC,RJ,RK,RI)              CCG22        58     A
SQZ                SA7    A5                                               CCG22        59     A
SQZ                EQ     SBB2                                             CCG22        60     A
SQZ                                                                       CCG22        61     A
SQZ       SBB5     UX7    "OC",X5                                          CCG22        62     A
SQZ                NZ     "OC",SBB2   IF OC[R1] " EOQ                      CCG22        63     A
SQZ                                                                       CCG22        64     A
SQZ                MX0    -R1.RIL           (X0) = R-NUMBER EXTRACTOR MASK CCG22        65     A
SQZ       .ROL     ENDIF                                                   CCG1        278     A
SQZ       SBB      EJECT                                                   SQZ         394     A
SQZ       **       SCAN *TXT* AND SETUP *RND*. CHAIN THE *RND* ENTRIES TO THE  SQZ     395     A

          0        1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ      *         INSTRUCTION THAT DEFINES THE *RI* ( RND INDEX ).                    SQZ        396    A
SQZ      *         SET PRECEDENCE BITS ( PS,PRS,RJRS ) FOR INSTRUCTIONS                SQZ        397    A
SQZ      *         PRECEDING A STORE / REGISTER STORE.                                 SQZ        398    A
SQZ                                                                                    SQZ        399    A
SQZ       SBB5A    SB4    B2+B2                                                        SQZ        400    A
SQZ                SA2    F.RDT+OC.RS       (X2) = DESCR FOR A *RJ* RS                 SQZ        401         I
          -CCG1
SQZ                SA2    =XF.RDT+OC.RS     (X2) = DESCRR FOR *RJ RS*                  CCG1       279         I
          -CCG18
SQZ                SA2    F.RDT+OC.RS       (X2) = DESCR FOR *RJ RS*                   CCG18      150    A
SQZ                SX7    B0          LINK = 0                                         SQZ        402    A
SQZ                SX1    B1                                                           SQZ        403    A
SQZ                LX1    D.^DP                                                        SQZ        404    A
SQZ                SB5    OC.RS                                                        SQZ        405    A
SQZ                BX2    -X1*X2                                                       SQZ        406    A
SQZ                LX1    D.PSP-D.^DP                                                  SQZ        407    A
SQZ                SB6    B0          RSL = 0                                          SQZ        408    A
SQZ                SA5    "TXT"                                                        SQZ        409    A
SQZ                EQ     SBB8                                                         SQZ        410    A
SQZ                                                                                    SQZ        411    A
SQZ       SBB6     SX6    B6                                                           SQZ        412    A
SQZ                SA6    A4+B1       [DI+1] = RSL                                     SQZ        413    A
SQZ                LX5    59-R1.INP                                                    SQZ        414    A
SQZ                SB6    A4+B1       RSL = DI + 1                                     SQZ        415    A
SQZ                PL     X5,SBB8     IF IN[R1] = 0  */ NORMAL REGISTER STORE          SQZ        416    A
SQZ                BX6    X2          ^D[DI] = 0                                       SQZ        417    A
SQZ                SA6    A4                                                           SQZ        418    A
SQZ                                                                                    SQZ        419    A
SQZ       SBB7     SX5    A5                                                           SQZ        420    A
SQZ                LX5    RN.TPP                                                       SQZ        421    A
SQZ                BX6    X5+X3       RND(RI) = RNDT(0,0,R1,RI[R1])                    SQZ        422    A
SQZ                SA6    "RN"+X3                                                      SQZ        423    A
SQZ                                                                                    SQZ        424    A
SQZ       SBB8     SA5    A5+B4       R1 = R1 + 4   */ ADVANCE TO NEXT INSTRUCTIO      SQZ        425    A
SQZ                SA4    A5+B2       DI = R1 + 2                                      SQZ        426    A
SQZ                BX3    -X0*X5                                                       SQZ        427    A
SQZ                LX4    59-D.^DP                                                     SQZ        428    A
SQZ                UX6    "OC",X5                                                      SQZ        429    A
SQZ                PL     X4,SBB7     IF D[DI]      */ DEFINES AN *RI* ( RESULT )      SQZ        430    A
SQZ                LX4    D.^DP-D.STP                                                  SQZ        431    A
SQZ                ZR     "OC",SBB9   IF OC[R1] = OC.EOQ                               SQZ        432    A
SQZ                EQ     "OC",B5,SBB6      IF OC[R1] = OC.RS                          SQZ        433    A
SQZ                PL     X4,SBB8     IF ^ST[DI]                                       SQZ        434    A
SQZ                IFNE   .DAL,0,2                                   LVL 2             SQZ        435         I
          -CCGA039
SQZ
>>>>      CCGA039 28                                                                   CCGA039     19    A
SQZ        #DAL    IFNE   .DAL,0
>>>>      CCGA039 29                                                                   CCGA039     20    A
SQZ                LX4    D.STP-D.L2P                                                  SQZ        436    A
SQZ                MI     X4,SBB8     IF L2[DI]                                        SQZ        437         I
          -CCGA039

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

SQZ                   PL     X4,SBB8A    IF ^L2[DI]
>>>>     CCGA039 31                                                    CCGA039       21     A
SQZ                   LX4    D.L2P-D.XMTP
>>>>     CCGA039 32                                                    CCGA039       22     A
SQZ                   PL     X4,SBB8     IF ^XMT[DI] */ LEVEL 2
>>>>     CCGA039 33                                                    CCGA039       23     A
SQZ
>>>>     CCGA039 34                                                    CCGA039       24     A
SQZ       SBB8A       BSS    0
>>>>     CCGA039 35                                                    CCGA039       25     A
SQZ       #DAL        ENDIF
>>>>     CCGA039 36                                                    CCGA039       26     A
SQZ
>>>>     CCGA039 37                                                    CCGA039       27     A
SQZ                   SA3    "RN"+X3                                  SQZ          438     A
SQZ                   SA7    A4+B1       [R1+3] = LINK                SQZ          439     A
SQZ                   LX3    -RN.TPP                                  SQZ          440     A
SQZ                   SX7    A7                                       SQZ          441     A
SQZ                   LX7    18          LINK = R1 + 3                SQZ          442     A
SQZ                   SA4    X3+B2       DJ = TP[RI] + 2              SQZ          443     A
SQZ                   BX6    X1+X4                                    SQZ          444     A
SQZ                   SA6    A4          PS[DJ] = 1                   SQZ          445     A
SQZ                   EQ     SBB8                                     SQZ          446     A
SQZ                                                                   SQZ          447     A
SQZ       SBB9        SX6    A5                                       SQZ          448           I
          -CCG17
SQZ                   SA6    EOQA        EOQA = R1     */ SAVE *EOQ* ADDRESS  SQZ  449           I
          -CCG17
SQZ                   LX7    -I.STCP     STC = LINK    */ ADDRESS OF FINAL STORE  SQZ  450       I
          -CCG17
SQZ       SBB9        LX7    -I.STCP     STC = LINK    */ ADDRESS OF A FINAL STORE  CCG17  7     A
SQZ                   SA7    STC                                      SQZ          451     A
SQZ                                                                   SQZ          452     A
SQZ       *           NOW SET THE REGISTER STORE PRECEDENCE BITS       SQZ          453     A
SQZ                                                                   SQZ          454     A
SQZ                   ZR     B6,SBB11    IF RSL = 0    */ NO REGISTER STORES  SQZ  455     A
SQZ                   LX1    D.PRSP-D.PSP                             SQZ          456     A
SQZ                   MX7    0                                        SQZ          457     A
SQZ                                                                   SQZ          458     A
SQZ       SBB10       SA2    B6-3        R1 = [RSL-3]                 SQZ          459     A
SQZ                   SA4    A2-B2       DI = R1 - 2   */ DESCR OF PREDECESSOR  SQZ  460     A
SQZ                   LX2    -R1.INP+1                                SQZ          461     A
SQZ                   SB5    X2                                       SQZ          462     A
SQZ                   SA2    B6                                       SQZ          463     A
SQZ                   LX5    B5,X1       BIT = SHIFT( PRS , 2*IN[R1] )  SQZ        464     A
SQZ                   SA7    B6          [R1+3] = 0                   SQZ          465     A
SQZ                   BX6    X5+X4       DI = DI ! BIT                SQZ          466     A
SQZ                   SB6    X2          RSL = [RSL]                  SQZ          467     A
SQZ                   SA6    A4                                       SQZ          468     A
SQZ                   NZ     B6,SBB10    IF RSL " 0                   SQZ          469     A
SQZ                                                                   SQZ          470     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

SQZ       *         COMPLETE INITIALIZATION OF *RND*, SET REGISTERS FOR MAIN LOOP          SQZ          471     A
SQZ                                                                                        SQZ          472     A
SQZ        SBB11    SX5     "TXT"                                                          SQZ          473     A
SQZ                 LX5     RN.TPP                                                         SQZ          474     A
SQZ                 MX6     1                                                              SQZ          475         I
          -CCG17
SQZ                 LX6     1+RN.ZP                                                        SQZ          476         I
          -CCG17
SQZ                 MX6     2                                                              CCG17          8     A
SQZ                 LX6     2+RN.CZP                                                       CCG17          9     A
SQZ                 SX4     B1                                                             SQZ          477     A
SQZ                 BX6     X5+X6                                                          SQZ          478     A
SQZ                 IX7     X5+X4                                                          SQZ          479     A
SQZ                 LX4     RN.USESP+3                                                     SQZ          480     A
SQZ                 BX6     X4+X6                                                          SQZ          481     A
SQZ                 SA6     "RN"        RND(0) = RNDT(Z,8,O.TXT,0)    ( B0 )               SQZ          482         I
          -CCG17
SQZ                 SA6     "RN"        RND(0) = RNDT(Z,C,8,O.TXT,0) ( B0 )                CCG17         10     A
SQZ                 SA7     "RN"+B1     RND(1) = RNDT(0,0,O.TXT,1)    ( A0 )               SQZ          483     A
SQZ                 CLEAR               CLEAR CHAIN POINTERS                               SQZ          484     A
SQZ                 SA6     RSS         RSS = 0                                            SQZ          485     A
SQZ                 SA6     JPXA        JPXA = 0                                           SQZ          486     A
SQZ                 SA6     IXF         IXF = 0                                            CCG17         11     A
SQZ                 SA4     "TXT"+B2    DI = O.TXT + 2                                     SQZ          487     A
SQZ                 MX6     1                                                              CCG17         12     A
SQZ                 SA6     LDCN        LDCN = 1S59   */ INDICATE NO LDC"S                 CCG17         13     A
SQZ                 TRACE   INITX,RLIST                                                    SQZ          488     A
SQZ                 EQ      MSL1                                                           SQZ          489         I
          -CCG17
SQZ        EOQ      SPACE   3                                                              SQZ          490         I
          -CCG17
SQZ       *         EOQ - END OF REDUCTION PHASE                                           SQZ          491         I
          -CCG17
SQZ                                                                                        SQZ          492         I
          -CCG17
SQZ                 PROCESS EOQ                                                            SQZ          493         I
          -CCG17
SQZ                 RJ      EDD         XFER USES COUNTS, MARK DEAD DEFINITIONS            SQZ          494         I
          -CCG17
SQZ                 TRACE   EDD,RLIST                                                      SQZ          495         I
          -CCG17
SQZ                 RJ      EDI         COMPRESS SEQUENCE                                  SQZ          496         I
          -CCG17
SQZ                 TRACE   SQZ.X,RLIST                                                    SQZ          497         I
          -CCG17
SQZ                                                                                        SQZ          498         I
          -CCG17
SQZ                 SA2     BIAS                                                           SQZ          499         I
          -CCG17
SQZ                 SA0     B5+                                                            SQZ          500         I
          -CCG17
SQZ                 CALL    RNI         RENUMBER THE INSTRUCTIONS                          SQZ          501         I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

                -CCGNC    -CCG17
      SQZ                 CALL   RNI#        RENUMBER THE INSTRUCTIONS                          CCGNC        761          I
                -CCG17
      SQZ                 SX6    0                                                             SQZ          502          I
                -CCG17
      SQZ                 SA6    L.RND       L.RND = 0                                         SQZ          503          I
                -CCG17
      SQZ                 EQ     SBB         RETURN                                            SQZ          504          I
                -CCG17
      SQZ        MSL      TITLE  MASTER SQUEEZE LOOP                                           SQZ          505     A
      SQZ       **        MSL - MASTER SQUEEZE LOOP                                            SQZ          506     A
      SQZ       *         REPLACE INSTRUCTIONS WHOSE RESULTS ARE KNOWN WITH SIMPLER            SQZ          507          I
                -CCG17
      SQZ       *         INSTRUCTIONS, OR ELIMINATE REDUNDANT INSTRUCTIONS                    SQZ          508          I
                -CCG17
      SQZ       *         MOST OF THE SPECIAL CASING HERE CENTERS ABOUT REDUCING               SQZ          509          I
                -CCG17
      SQZ       *         INSTRUCTIONS WHOSE OPERANDS ARE ZERO.  THIS IS NECESSARY             SQZ          510          I
                -CCG17
      SQZ       *         BECAUSE IN PASS 1, *ARITH* OUTPUTS MIXED MODE COMPUTATIONS AS        SQZ          511          I
                -CCG17
      SQZ       *         (REAL,0) OP COMPLEX , ETC.                                           SQZ          512          I
                -CCG17
      SQZ       *         EVALUATE INSTRUCTIONS WITH CONSTANT OPERANDS OR REDUCE TO            CCG17         14     A
      SQZ       *         SIMPLER INSTRUCTION BY AN ALGEBRAIC IDENTITY, IF ONE OF THE          CCG17         15     A
      SQZ       *         OPERANDS IS CONSTANT. ALSO ATTEMPT TO ELIMINATE ANY                  CCG17         16     A
      SQZ       *         REDUNDANT INSTRUCTIONS.                                              CCG17         17     A
      SQZ                                                                                      SQZ          513     A
      SQZ                                                                                      SQZ          514     A
      SQZ                 PROCESS EOS                                                          SQZ          515          I
                -FCC0588
      SQZ        MSL1     SA5    A4+B2       R1 = DI + 2   */ ADVANCE TO NEXT INSTRUCTIO       SQZ          516     A
      SQZ                 SA4    A5+B2       DI = R1 + 2                                       SQZ          517     A
      SQZ                 MX1    D.TYL                                                         SQZ          518     A
      SQZ                 LX4    58-D.TYP                                                      SQZ          519     A
      SQZ                 BX6    X1*X4                                                         SQZ          520     A
      SQZ                 LX4    D.TYP+2                                                       SQZ          521     A
      SQZ                 UX7    "OC",X5                                                       SQZ          522     A
      SQZ                 NZ     X6,MSL3     IF TYPE[DI] " I                                   SQZ          523     A
      SQZ                                                                                      SQZ          524     A
      SQZ       *         TYPE I - REPLACE OPERAND R-NUMBERS WITH EQUIVALENTS FROM RND.        SQZ          525     A
      SQZ                                                                                      SQZ          526     A
      SQZ                 LX7    -R1.RKP                                                       SQZ          527     A
      SQZ                 BX6    -X0*X7                                                        SQZ          528     A
      SQZ                 SA2    "RN"+X6     (X2) = RK = RND(RK)                               SQZ          529     A
      SQZ                 LX7    R1.RKP-R1.RJP                                                 SQZ          530     A
      SQZ                 SA1    "RN"+X7     (X1) = RJ = RND(RJ)                               SQZ          531     A
      SQZ                 LX4    59-D.CMP                                                      SQZ          532     A
      SQZ                 BX5    -X0*X5                                                        SQZ          533     A
      SQZ                 PL     X4,MSL2     IF ^CM[DI]                                        SQZ          534     A
      SQZ                 SB4    X1                                                            SQZ          535     A
      SQZ                 SB5    X2                                                            SQZ          536     A

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                GE     B4,B5,MSL2  IF RJ \ RK    */ CANONICAL ORDER                SQZ          537     A
SQZ                SA1    A2+                                                           SQZ          538     A
SQZ                SA2    "RN"+X7     SWAP( RJ , RK )                                   SQZ          539     A
SQZ                                                                                     SQZ          540     A
SQZ        MSL2    BX3    -X0*X1                                                        SQZ          541     A
SQZ                LX4    1+D.CMP                                                       SQZ          542     A
SQZ                PX5    "OC",X5                                                       SQZ          543     A
SQZ                BX6    -X0*X2                                                        SQZ          544     A
SQZ                LX3    R1.RJP                                                        SQZ          545     A
SQZ                BX5    X3+X5                                                         SQZ          546     A
SQZ                LX6    R1.RKP                                                        SQZ          547     A
SQZ                IX7    X5+X6                                                         SQZ          548     A
SQZ                BX5    X5+X6                                                         SQZ          549     A
SQZ                SA7    A5+0        [R1] = TYI(OC,RJ,RK,RI)                           SQZ          550         I
           -CCG17
SQZ                                                                                     SQZ          551         I
           -CCG17
SQZ                BX6    X1*X2                                                         CCG17         18     A
SQZ                SA7    A5          [R1] = TYI(OC RJ,RK,RI)                           CCG17         19     A
SQZ                LX6    59-RN.CP                                                      CCG17         20     A
SQZ                PL     X6,MSL3     IF ^(C[RJ] & C[RK])                               CCG17         21     A
SQZ                                                                                     CCG17         22     A
SQZ        *       TYPE I AND CONSTANT OPERANDS, JUMP TO SPECIAL PROCESSOR TO           CCG17         23     A
SQZ        *       EVALUATE THE VALUE OF THIS INSTRUCTION.                              CCG17         24     A
SQZ                                                                                     CCG17         25     A
SQZ                SA3    SBB.JT+"OC"                                                   CCG17         26     A
SQZ                BX6    -X0*X5                                                        CCG17         27     A
SQZ                SB6    X6          (B6) = RI                                         CCG17         28     A
SQZ                SB5    X3+                                                           CCG17         29     A
SQZ                JP     B5                                                            CCG17         30     A
SQZ        MSL     SPACE  2,10                                                          CCG17         31     A
SQZ        **      JUMP TO INSTRUCTION PROCESSORS WITH -                                SQZ          552     A
SQZ        *       (A5) = R1 = ADDRESS OF R1 WORD  , (X5) = [R1]                        SQZ          553     A
SQZ        *       (A4) = DI, ADDRESS OF DESCRIPTOR , (X4) = [DI]                       SQZ          554     A
SQZ        *       (B6) = RI = RI[R1]                                                   SQZ          555     A
SQZ        *       FOR TYPE I INSTRUCTIONS, (X1/X2) = RND(RJ)/RND(RK)                   SQZ          556     A
SQZ                                                                                     SQZ          557     A
SQZ        MSL3    BX3    -X0*X5                                                        SQZ          558         I
           -CCG17
SQZ                SB6    X3          (B6) = RI                                         SQZ          559         I
           -CCG17
SQZ        MSL3    BX6    -X0*X5                                                        CCG17         32     A
SQZ                SB6    X6          (B6) = RI                                         CCG17         33     A
SQZ                JP     SBB.JT+"OC" JUMP( SBB.JT(OC[R1]) )                            SQZ          560     A
SQZ        EOQ     SPACE  3,14                                                          CCG17         34     A
SQZ        *       EOQ - END OF SEQUENCE                                                CCG17         35     A
SQZ        *       MAKE FURTHER PASS"S OVER THE INSTRUCTIONS TO ELIMINATE               CCG17         36     A
SQZ        *       THE USELESS DEFINITIONS AND COMPRESS OUT THE DEAD INSTRUCTIONS.      CCG17         37     A
SQZ                                                                                     CCG17         38     A
SQZ                PROCESS EOQ                                                          CCG17         39     A
SQZ                RJ     EDD         BACKWARDS SCAN TO TRANSFER USES                   CCG17         40     A
SQZ        *                          AND ELIMINATE USELESS DEFINITIONS                 CCG17         41     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                RJ     UCT         UPDATE CVT/CUT                                  CCG17        42     A
SQZ                TRACE  EDD,RLIST                                                   CCG17        43     A
SQZ                                                                                   CCG17        44     A
SQZ                RJ     EDI         COMPRESS SEQUENCE                               CCG17        45     A
SQZ                                                                                   CCG17        46     A
SQZ                SA2    BIAS                                                        CCG17        47     A
SQZ                SA0    B5+                                                         CCG17        48     A
SQZ                CALL   RNI#        RENUMBER THE INSTRUCTIONS                       CCG17        49     A
SQZ                SX6    0                                                           CCG17        50     A
SQZ                SA6    L.RND       L.RND = 0                                       CCG17        51     A
SQZ                TRACE  SQZ.X,RLIST                                                 CCG17        52     A
SQZ                                                                                   CCG17        53     A
SQZ                RJ     IXI         INSERT EXTRA INSTRUCTIONS                       CCG17        54     A
SQZ                EQ     SBB                                                         CCG17        55     A
SQZ       MSL      SPACE  3,14                                                        CCG17        56     A
SQZ      **        EPROC - DEFINE ADDRESSES FOR NAMES OF EVALUATE                     CCG17        57     A
SQZ      *                   CONSTANT PROCESSORS                                     CCG17        58     A
SQZ      *                                                                           CCG17        59     A
SQZ      *         EPROC  (A,B,C,...,Z)                                               CCG17        60     A
SQZ                                                                                   CCG17        61     A
SQZ                PURGMAC EPROC                                                      CCG17        62     A
SQZ       EPROC    MACRO  P                                                           CCG17        63     A
SQZ                IRP    P                                                           CCG17        64     A
SQZ                NOREF  )P                                                          CCG17        65     A
SQZ       )P       BSS    0                                                           CCG17        66     A
SQZ                IRP                                                                CCG17        67     A
SQZ       EPROC    ENDM                                                               CCG17        68     A
SQZ                                                                                   CCG17        69     A
SQZ       SXT      TITLE  MASTER SQUEEZE LOOP - TYPE I INSTRUCTIONS                   SQZ         561     A
SQZ      **        SPECIAL PROCESSING OF TYPE I INSTRUCTIONS                          SQZ         562     A
SQZ                                                                                   SQZ         563     A
SQZ                PROCESS SXT        CONSTANT SHIFT PREDECESSOR                      SQZ         564     A
SQZ                EPROC  SXT                                                         CCG17        70     A
SQZ                SA3    A4+4        DIS = DI + 4                                    SQZ         565     A
SQZ                LX3    59-D.KSP                                                    SQZ         566     A
SQZ                PL     X3,AUC.RJ   IF ^KS[DIS]   */ NEXT IS NOT A CON SHIFT       SQZ         567     A
SQZ                LX1    59-RN.CP                                                    SQZ         568     A
SQZ                PL     X1,SXT0     IF ^C[RJ]                                       SQZ         569     A
SQZ                                                                                   SQZ         570     A
SQZ      *         CONSTANT OPERAND, EVALUATE SXT, SHIFT COMBINATION                  SQZ         571     A
SQZ                                                                                   SQZ         572     A
SQZ                LX1    1+RN.CP-RN.TPP                                              SQZ         573     A
SQZ                SA3    X1          R1J = TP[RJ]                                    SQZ         574     A
SQZ                RJ     DCV         VAL = DCV(R1J)                                  SQZ         575     A
SQZ                SA2    A4+B2       R1S = DI + 2                                    SQZ         576     A
SQZ                SA3    A2+B1       R2S = R1S + 1                                   SQZ         577     A
SQZ                UX2    B3,X2                                                       SQZ         578     A
SQZ                LX3    -IH.CAP                                                     SQZ         579     A
SQZ                SB3    B3-OC.KLS                                                   SQZ         580     A
SQZ                LX1    RN.TPP+59-RN.CP                                             SQZ         581     A
SQZ                SB4    X3                                                          SQZ         582     A
SQZ      +         LX6    B4,X7                                                       SQZ         583     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                ZR     B3,*+1      VAL[RI] = SHIFT( VAL[RJ] , CA[R2S] )          SQZ        584    A
SQZ                AX6    B4,X7                                                     SQZ        585    A
SQZ                SX3    X6                                                        SQZ        586         I
          -CCG17
SQZ                BX7    X6-X3                                                     SQZ        587         I
          -CCG17
SQZ                NZ     X7,SXT0     IF VAL " SXIXJ(VAL)                           SQZ        588         I
          -CCG17
SQZ                                                                                SQZ        589         I
          -CCG17
SQZ                MX7    0                                                         CCG17       71    A
SQZ                SA7    A4          [DI] = 0      */ KILL THE *SXT*               SQZ        590    A
SQZ                SA5    A4+B2       R1 = DI + 2                                   SQZ        591    A
SQZ                SA4    A5+B2       DI = R1 + 2   */ ADVANCE TO NEXT INSTRUCTION  SQZ        592    A
SQZ                BX3    -X0*X5                                                    SQZ        593    A
SQZ                SB6    X3                                                        SQZ        594    A
SQZ                EQ     CCV                                                       SQZ        595    A
SQZ                                                                                SQZ        596    A
SQZ       SXT0     SA2    SXTC                                                      SQZ        597    A
SQZ                ZR     X2,SXT4     IF SXTC = 0   */ FIRST *SXT*                  SQZ        598    A
SQZ                LX1    1+RN.CP-RN.TPP                                            SQZ        599    A
SQZ                SB5    X1          DEFA = TP[RJ]                                 SQZ        600    A
SQZ                BX5    X0*X5       KEY = TYI(OC,RJ,0,0)                          SQZ        601    A
SQZ                SA3    A4+B2                                                     SQZ        602    A
SQZ                UX7    B4,X3                                                     SQZ        603    A
SQZ                SA3    A3+B1                                                     SQZ        604    A
SQZ                LX3    -IH.CAP                                                   SQZ        605    A
SQZ                BX6    -X0*X3                                                    SQZ        606    A
SQZ                PX4    B4,X6       OC.CA = TY1(OC[R1+4],0,0,CA[R1+5])            SQZ        607    A
SQZ                EQ     SXT2                                                      SQZ        608    A
SQZ                                                                                SQZ        609    A
SQZ       *        SEARCH THE SXT CHAIN FOR A PREVIOUS OCCURRENCE WITH THE           SQZ        610    A
SQZ       *        SAME SHIFT FOLLOWING                                              SQZ        611    A
SQZ                                                                                SQZ        612    A
SQZ       SXT1     SA2    X2          SXTC = [SXTC]                                 SQZ        613    A
SQZ                SB4    X2                                                        SQZ        614    A
SQZ                LE     B4,B5,SXT3  IF SXTC @ DEFA                                SQZ        615    A
SQZ       SXT2     SA3    X2-3        R1C = [SXTC-3]                                SQZ        616    A
SQZ                BX6    X0*X3                                                     SQZ        617    A
SQZ                IX7    X5-X6                                                     SQZ        618    A
SQZ                NZ     X7,SXT1     IF KEY " TYI(OC,RJ,0,0)(SXTC)                 SQZ        619    A
SQZ                SA3    A3+5                                                      SQZ        620    A
SQZ                LX3    -IH.CAP                                                   SQZ        621    A
SQZ                BX7    -X0*X3                                                    SQZ        622    A
SQZ                SA3    A3-B1                                                     SQZ        623    A
SQZ                UX6    B4,X3                                                     SQZ        624    A
SQZ                PX6    B4,X7                                                     SQZ        625    A
SQZ                IX7    X6-X4                                                     SQZ        626    A
SQZ                NZ     X7,SXT1     IF OC.CA " OC.CA(SXTC+4)                      SQZ        627    A
SQZ                                                                                SQZ        628    A
SQZ       *        MARK THE *SXT* AND THE FOLLOWING CON SHIFT AS REDUNDANT           SQZ        629    A
SQZ       *        NOTE - CON SHIFTS MAY NOT PRECEDE A STORE / REGISTER STORE.       SQZ        630    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                                                                                  SQZ          631     A
SQZ                SA7    A4          [DI] = 0                                       SQZ          632     A
SQZ                BX2    -X0*X3      RIP = RI[R1C]   */ RI OF FIRST SHIFT           SQZ          633     A
SQZ                SA5    A4+B2       R1 = DI + 2   */ ADVANCE TO NEXT INST          SQZ          634     A
SQZ                SA7    A5+B2       [R1+2] = 0    */ MARK SHIFT DEAD               SQZ          635     A
SQZ                BX4    -X0*X5                                                     SQZ          636     A
SQZ                SA3    "RN"+X2                                                    SQZ          637     A
SQZ                BX7    X3                                                         SQZ          638     A
SQZ                SA7    "RN"+X4     RND(RI) = RND(RIP)                             SQZ          639     A
SQZ                SA4    A5+B2       DI = R1 + 2                                    SQZ          640     A
SQZ                EQ     MSL1                                                       SQZ          641     A
SQZ                                                                                  SQZ          642     A
SQZ      *         ADD THE *SXT* TO THE CHAIN                                        SQZ          643     A
SQZ                                                                                  SQZ          644     A
SQZ       SXT3     SA2    SXTC                                                       SQZ          645     A
SQZ                BX7    X2                                                         SQZ          646     A
SQZ                SA7    A4+B1       [DI+1] = SXTC                                  SQZ          647     A
SQZ       SXT4     SX6    A4+B1                                                      SQZ          648     A
SQZ                SA6    A2          SXTC = DI + 1                                  SQZ          649     A
SQZ      *         EQ     AUC.RJ                                                     SQZ          650     A
SQZ       AUC.RJ   SPACE  3                                                          SQZ          651     A
SQZ      **        AUC.RJ - ADVANCE USES COUNT FOR R-NUMBER IN *RJ* FIELD            SQZ          652     A
SQZ                                                                                  SQZ          653     A
SQZ                PROCESS DAR                                                       SQZ          654     A
SQZ       AUC.RJ   LX5    -R1.RJP                                                    SQZ          655     A
SQZ                SX2    B1                                                         SQZ          656     A
SQZ                BX3    -X0*X5                                                     SQZ          657     A
SQZ                SA1    "RN"+X3                                                    SQZ          658     A
SQZ                LX2    RN.USESP                                                   SQZ          659     A
SQZ                IX6    X2+X1       USES[RJ] = USES[RJ] + 1                        SQZ          660     A
SQZ                SA6    A1+                                                        SQZ          661     A
SQZ                EQ     MSL1                                                       SQZ          662     A
SQZ                                                                                  SQZ          663         I
          -CCG17                                                                                                 I
SQZ       KLS      SPACE  2,14                                                       CCG17         72     A
SQZ                PROCESS (KLS,KRS)  NON REDUNDANT CON SHIFT                        SQZ          664     A
SQZ                SA2    A5+B1       R2 = R1 + 1                                    SQZ          665     A
SQZ                SX3    B1                                                         SQZ          666         I
          -CCG17                                                                                                 I
SQZ                LX2    -IH.RFP                                                    SQZ          667         I
          -CCG17                                                                                                 I
SQZ                LX2    -IH.CAP                                                    CCG17         73     A
SQZ                SB4    X2          CS = CA[R2]                                    CCG17         74     A
SQZ                SB5    "OC"-OC.KLS                                                CCG17         75     A
SQZ                ZR     B4,KLS1     IF CS = 0                                      CCG17         76     A
SQZ                NZ     B5,KLS2     IF OC " KLS                                    CCG17         77     A
SQZ                SB5    B4-60                                                      CCG17         78     A
SQZ                NZ     B5,KLS2     IF OC " 60                                     CCG17         79     A
SQZ                                                                                  CCG17         80     A
SQZ       KLS1     MX6    0                                                          CCG17         81     A
SQZ                SA6    A4-4        DI[SXT] = 0   */ KILL SXT INSTR.               CCG17         82     A
SQZ                SA2    A6-B2       R1S = DI[SXT] - 2                              CCG17         83     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                LX2    -R1.RJP                                          CCG17        84     A
SQZ                BX6    -X0*X2       RJS = RJ[R1S]                       CCG17        85     A
SQZ                SA1    "RN"+X6      (X1) = RND(RJS)                     CCG17        86     A
SQZ                SX7    B1                                               CCG40        12     A
SQZ                LX7    RN.USESP                                         CCG40        13     A
SQZ                IX7    X1-X7                                            CCG40        14     A
SQZ                SA7    A1           DECREMENT USES COUNT ON SHIFT PRED. CCG40        15     A
SQZ                BX1    X7                                               CCG40        16     A
SQZ                LX6    R1.RJP                                           CCG17        87     A
SQZ                EQ     XMT0         GO TO GENERATE  (XMT RI,RJS )       CCG17        88     A
SQZ                                                                       CCG17        89     A
SQZ        KLS2    LX2    IH.CAP-IH.RFP                                    CCG17        90     A
SQZ                SX3    B1                                               CCG17        91     A
SQZ                SA1    "RN"+X2      RF = RND(RF)                        SQZ         668     A
SQZ                LX3    RN.USESP                                         SQZ         669     A
SQZ                IX6    X3+X1        USES[RF] = USES[RF] + 1             SQZ         670     A
SQZ                SA6    A1                                               SQZ         671     A
SQZ                EQ     MSL1                                             SQZ         672     A
SQZ        XMT     SPACE  3,14                                             SQZ         673     A
SQZ        XMT0    S"OC"  OC.XMT                                           SQZ         674     A
SQZ                RJ     URI          GENERATE AN *XMT*                   SQZ         675     A
SQZ                                                                       SQZ         676     A
SQZ                EPROC  XMT                                              CCG17        92     A
SQZ        XMT     PROCESS XMT                                            SQZ         677     A
SQZ                RJ     CTI          CHECK TRANSMIT INSTRUCTION          SQZ         678     A
SQZ                EQ     SRCH                                             SQZ         679           I
           -FCC0603
SQZ                BX6    X4                                               FCC0603      89     A
SQZ                LX6    59-D.PRSP                                        FCC0603      90     A
SQZ                PL     X6,SRCH      IF ^PRS[DI]                         FCC0603      91     A
SQZ                EQ     AUC.RJ                                           FCC0603      92     A
SQZ         XMTC   SPACE  2                                                SQZ         680     A
SQZ       *        XMTC - UNARY COMPLEMENT                                 SQZ         681     A
SQZ                                                                       SQZ         682     A
SQZ                EPROC  XMTC                                             CCG17        93     A
SQZ         XMTC   PROCESS XMTC                                            SQZ         683     A
SQZ                BX6    X1                                               SQZ         684           I
           -CCG17
SQZ                LX6    59-RN.CP                                         SQZ         685           I
           -CCG17
SQZ                MI     X6,XMTC1     IF C[RJ]                            SQZ         686           I
           -CCG17
SQZ                LX6    RN.CP-RN.XMTCP                                   SQZ         687           I
           -CCG17
SQZ                PL     X6,XMTC2     IF ^XMTC[RJ]                        SQZ         688           I
           -CCG17
SQZ                BX6    X1                                               CCG17        94     A
SQZ                LX6    59-RN.XMTCP                                      CCG17        95     A
SQZ                PL     X6,XMTC1                                         CCG17        96     A
SQZ                                                                       CCG17        97     A
SQZ       *        REDUCE -(-X) TO X                                       CCG17        98     A
SQZ                                                                       SQZ         689     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                LX1    -RN.TPP                                          SQZ          690     A
SQZ                SA2    X1          R1P = R1[TP[RJ]]                     SQZ          691     A
SQZ                LX2    -R1.RJP                                          SQZ          692     A
SQZ                BX6    -X0*X2                                           SQZ          693     A
SQZ                SA1    "RN"+X6                                          SQZ          694     A
SQZ                LX6    R1.RJP                                           SQZ          695     A
SQZ                EQ     XMT0                                             SQZ          696     A
SQZ                                                                        SQZ          697     A
SQZ       XMTC1    LX1    -RN.TPP                                          SQZ          698         I
         -CCG17
SQZ                SA3    X1          R1J = TP[RJ]                         SQZ          699         I
         -CCG17
SQZ                RJ     DCV         VAL = DCV(R1J)                       SQZ          700         I
         -CCG17
SQZ                BX6    -X7         VAL = -VAL                           SQZ          701         I
         -CCG17
SQZ                MI     X6,CCV      IF VAL < 0                           SQZ          702         I
         -CCG17
SQZ                SX3    X6                                               SQZ          703         I
         -CCG17
SQZ                BX7    X6-X3                                            SQZ          704         I
         -CCG17
SQZ                NZ     X7,XMTC2    IF SETX(VAL) " VAL                   SQZ          705         I
         -CCG17
SQZ                ZR     B5,CCV      IF OC[RJ] = OC.S                     SQZ          706         I
         -CCG17
SQZ                LX1    RN.TPP                                           SQZ          707         I
         -CCG17
SQZ                SA3    "RN"+B6                                          SQZ          708         I
         -CCG17
SQZ                BX7    -X0*X1                                           SQZ          709         I
         -CCG17
SQZ                SX2    B1                                               SQZ          710         I
         -CCG17
SQZ                SA7    A5+B1       [R2] = RJ[R1J]  */ POINT TO RJ FOR *AND*  SQZ          711         I
         -CCG17
SQZ                LX2    RN.FMAP                                          SQZ          712         I
         -CCG17
SQZ                BX7    X2+X3       FMA[RI] = 1                          SQZ          713         I
         -CCG17
SQZ                SA7    A3                                               SQZ          714         I
         -CCG17
SQZ                LX6    R1.INP                                           SQZ          715         I
         -CCG17
SQZ                S"OC"  OC.S                                             SQZ          716         I
         -CCG17
SQZ                RJ     URI         URI(  S  RI,VAL )                    SQZ          717         I
         -CCG17
SQZ                EQ     FMA                                              SQZ          718         I
         -CCG17
SQZ                                                                        SQZ          719         I
         -CCG17

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ       XMTC2    SA2    "RN"+B6                                          SQZ          720          I
         -CCG17
SQZ       XMTC1    LX1    59-RN.CP                                         CCG17         99          I
         -CCGA047
SQZ                PL     X1,XMTC2    IF ^C[RI]                            CCG17        100          I
         -CCGA047
SQZ                                                                       CCG17        101          I
         -CCGA047
SQZ                LX1    1+RN.CP-RN.TPP                                   CCG17        102          I
         -CCGA047
SQZ       XMTC1    BX6    X1                                               CCGA047        6     A
SQZ                LX6    59-RN.CP                                         CCGA047        7     A
SQZ                PL     X6,XMTC2    IF ^C[RI]                            CCGA047        8     A
SQZ                                                                       CCGA047        9     A
SQZ                LX1    -RN.TPP                                          CCGA047       10     A
SQZ                SA3    X1                                               CCG17        103     A
SQZ                RJ     DCV                                              CCG17        104     A
SQZ                BX6    -X7         VAL = -DCV(TP[RI])                   CCG17        105     A
SQZ                LX1    RN.TPP                                           CCG17        106     A
SQZ                SB4    OC.FMA                                           CCG17        107     A
SQZ                EQ     B5,B4,XMTC2 IF OC = OC.FMA  */ NO REDUCTION      CCG17        108     A
SQZ                SB4    OC.LDC                                           CCG17        109     A
SQZ                NE     B5,B4,CCV   IF OC " OC.LDC                       CCG17        110     A
SQZ                BX7    X4                                               CCG17        111     A
SQZ                LX7    59-D.PSP                                         CCG17        112     A
SQZ                PL     X7,CCV      IF ^PS[DI]                           CCG17        113     A
SQZ                                                                       CCG17        114     A
SQZ       XMTC2    SA2    "RN"+B6                                          CCG17        115     A
SQZ                SX6    B1                                               SQZ          721     A
SQZ                LX6    RN.XMTCP                                         SQZ          722     A
SQZ                BX7    X6+X2       XMTC[RI] = 1                         SQZ          723     A
SQZ                LX6    RN.CP-RN.XMTCP                                   CCG17        116     A
SQZ                BX1    X6*X1                                            CCG17        117     A
SQZ                BX7    X7+X1       C[RI] = C[RJ]                        CCG17        118     A
SQZ                SA7    A2                                               SQZ          724     A
SQZ                EQ     SRCH                                             SQZ          725     A
SQZ       AND      EJECT                                                   SQZ          726     A
SQZ                PROCESS AND                                             SQZ          727     A
SQZ                BX6    X1+X2                                            SQZ          728     A
SQZ                LX6    59-RN.ZP                                         SQZ          729     A
SQZ                MI     X6,ZERO     IF Z[RJ] ! Z[RK]                     SQZ          730     A
SQZ                IX7    X1-X2                                            SQZ          731     A
SQZ                ZR     X7,C.XMT    IF RJ = RK                           SQZ          732     A
SQZ                LX6    RN.ZP-RN.CP                                      SQZ          733          I
         -CCG17
SQZ                PL     X6,AND3     IF ^( C[RJ] ! C[RK] )                SQZ          734          I
         -CCG17
SQZ                BX7    X1*X2                                            SQZ          735          I
         -CCG17
SQZ                LX7    59-RN.CP                                         SQZ          736          I
         -CCG17
SQZ                MI     X7,ECO      IF C[RJ] & C[RK]                     SQZ          737          I

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
          -CCG17
SQZ                                                                                  SQZ        738        I
          -CCG17
SQZ       *         SEE IF CON IS A *FMA* CHANGED TO A *S*                             SQZ        739        I
          -CCG17
SQZ       *         IF SO, THEN CHANGE TO STR( FMA , RK )                              SQZ        740        I
          -CCG17
SQZ                                                                                  SQZ        741        I
          -CCG17
SQZ                 BX7    X6                                                          SQZ        742        I
          -CCG17
SQZ                 LX7    RN.CP-RN.FMAP                                               SQZ        743        I
          -CCG17
SQZ                 PL     X7,AND3     IF ^(FMA[RJ] ! FMA[RK] )                        SQZ        744        I
          -CCG17
SQZ                                                                                  SQZ        745        I
          -CCG17
SQZ                 BX6    X1                                                          SQZ        746        I
          -CCG17
SQZ                 LX6    59-RN.FMAP                                                  SQZ        747        I
          -CCG17
SQZ                 MI     X6,AND2     IF FMA[RJ]                                      SQZ        748        I
          -CCG17
SQZ                 BX1    X2                                                          SQZ        749        I
          -CCG17
SQZ                 LX6    1+RN.FMAP   SWAP( RJ , RK )                                 SQZ        750        I
          -CCG17
SQZ                 BX2    X6                                                          SQZ        751        I
          -CCG17
SQZ                                                                                  SQZ        752        I
          -CCG17
SQZ        AND2     LX1    -RN.TPP                                                     SQZ        753        I
          -CCG17
SQZ                 BX2    -X0*X2                                                      SQZ        754        I
          -CCG17
SQZ                 SA1    X1+B1       RJ = [TP[RJ]+1]  */ R2 OF *S*                   SQZ        755        I
          -CCG17
SQZ                 LX2    R1.RKP                                                      SQZ        756        I
          -CCG17
SQZ                 S"OC"  OC.STR                                                      SQZ        757        I
          -CCG17
SQZ                 LX1    R1.RJP                                                      SQZ        758        I
          -CCG17
SQZ                 BX6    X1+X2                                                       SQZ        759        I
          -CCG17
SQZ                 EQ     CAO2                                                        SQZ        760        I
          -CCG17
SQZ                                                                                  SQZ        761        I
          -CCG17
SQZ        AND3     LX6    RN.CP-RN.XMTCP                                              SQZ        762        I
          -CCG17
SQZ                 LX6    RN.ZP-RN.XMTCP                                              CCG17      119      A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                PL     X6,SRCH     IF ^XMTC[RJ] & ^XMTC[RK]                    SQZ          763     A
SQZ                S"OC"  OC.STR                                                  SQZ          764     A
SQZ                SB5    R1.RJP                                                  SQZ          765     A
SQZ                EQ     CAO                                                     SQZ          766     A
SQZ         DFM    EJECT                                                          SQZ          767          I
           -CCG17
SQZ                PROCESS (DFM,IM)                                               SQZ          768          I
           -CCG17
SQZ                BX6    X1+X2                                                   SQZ          769          I
           -CCG17
SQZ                LX6    59-RN.ZP                                                SQZ          770          I
           -CCG17
SQZ                MI     X6,ZERO     IF Z[RJ] ! Z[RK]                            SQZ          771          I
           -CCG17
SQZ                LX6    RN.ZP-RN.CP                                             SQZ          772          I
           -CCG17
SQZ         DFM    SPACE  3,14                                                   CCG17        120     A
SQZ        *       MULTIPLY INSTRUCTION PROCESSING                               CCG17        121     A
SQZ                                                                              CCG17        122     A
SQZ                PROCESS IM                                                    CCG17        123     A
SQZ                BX6    X4                                                      CCG17        124     A
SQZ                LX6    59-D.USIP                                               CCG17        125     A
SQZ                PL     X6,DFM      IF ^USI[DI]                                 CCG17        126     A
SQZ                SX6    1                                                       CCG17        127     A
SQZ                SA6    IXF         IXF = 1       */ SET FOR *IXI* SCAN         CCG17        128     A
SQZ                                                                              CCG17        129     A
SQZ         DFM    PROCESS (DFM,FM,RFM)                                          CCG17        130     A
SQZ                BX6    X1+X2                                                   CCG17        131     A
SQZ                LX6    59-RN.CP                                                CCG17        132     A
SQZ                PL     X6,SRCH     IF ^( C[RJ] ! C[RK] )                       SQZ          773     A
SQZ                BX7    X1*X2                                                   SQZ          774          I
           -CCG17
SQZ                LX7    59-RN.CP                                                SQZ          775          I
           -CCG17
SQZ                PL     X7,DFM1     IF ^( C[RJ] & C[RK] )                       SQZ          776          I
           -CCG17
SQZ                                                                              SQZ          777          I
           -CCG17
SQZ                S"OC"  OC.DFM                                                  SQZ          778          I
           -CCG17
SQZ                EQ     ECO                                                     SQZ          779          I
           -CCG17
SQZ                                                                              SQZ          780          I
           -CCG17
SQZ         DFM1   BX7    X1                                                      SQZ          781          I
           -CCG17
SQZ                LX7    59-RN.CP                                                SQZ          782          I
           -CCG17
SQZ                MI     X7,DFM2     IF C[RJ]                                    SQZ          783          I
           -CCG17
SQZ                BX7    X1                                                      CCG17        133     A
SQZ                LX7    59-RN.CP                                                CCG17        134     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                MI     X7,DFM1     IF C[RJ]                                  CCG17        135     A
SQZ                BX1    X2                                                    SQZ          784     A
SQZ                LX7    1+RN.CP                                               SQZ          785     A
SQZ                BX2    X7                                                    SQZ          786     A
SQZ                                                                             SQZ          787     A
SQZ       DFM2     LX1    -RN.TPP                                               SQZ          788         I
         -CCG17
SQZ       DFM1     LX1    -RN.TPP                                               CCG17        136     A
SQZ                SA3    X1                                                    SQZ          789     A
SQZ                RJ     DCV         VAL = VAL[RJ/RK]                          SQZ          790     A
SQZ                MX6    59                                                    SQZ          791         I
         -CCG17
SQZ                IX3    X7-X6                                                 SQZ          792         I
         -CCG17
SQZ                ZR     X7,ZERO     IF VAL = -0                               SQZ          793         I
         -CCG17
SQZ                ZR     X7,ZERO     IF VAL = 0                                CCG17        137     A
SQZ                SB4    "OC"-OC.IM                                            CCG17        138     A
SQZ                NZ     B4,FM1      IF OC " IM                                CCG17        139     A
SQZ                SX3    B1                                                    CCG17        140     A
SQZ                IX3    X7+X3                                                 CCG17        141     A
SQZ                MI     X3,SRCH     IF VAL < -1                               SQZ          794     A
SQZ                AX3    2                                                     SQZ          795     A
SQZ                SB4    X7                                                    SQZ          796     A
SQZ                NZ     X3,SRCH     IF VAL > 2                                SQZ          797     A
SQZ                SA1    "RN"+X2                                               SQZ          798     A
SQZ                BX6    -X0*X2                                                SQZ          799     A
SQZ                LX6    R1.RJP                                                SQZ          800     A
SQZ                EQ     B4,B2,DFM3  IF VAL = 2                                SQZ          801         I
         -CCG17
SQZ                S"OC"  OC.IA                                                 CCG17        142     A
SQZ                EQ     B4,B2,DFM3  IF VAL = 2                                CCG17        143     A
SQZ                EQ     B4,B1,XMT0  IF VAL = 1                                SQZ          802     A
SQZ                S"OC"  OC.XMTC                                               SQZ          803         I
         -CCG17
SQZ                                                                             CCG17        144     A
SQZ      *         VAL = -1                                                    CCG17        145     A
SQZ                                                                             CCG17        146     A
SQZ       DFM2     S"OC"  OC.XMTC                                               CCG17        147     A
SQZ                RJ     URI         URI(  XMTC  RI,RJ )                       SQZ          804     A
SQZ                EQ     XMTC                                                  SQZ          805     A
SQZ                                                                             SQZ          806     A
SQZ       DFM3     BX7    X6                                                    SQZ          807         I
         -CCG17
SQZ                                                                             CCG17        148     A
SQZ      *         CONVERT 2*X TO X+X                                          CCG17        149     A
SQZ                                                                             CCG17        150     A
SQZ       DFM4     SB4    "OC"-OC.FM                                            CCG22         66     A
SQZ                S"OC"  OC.FA                                                 CCG22         67     A
SQZ                ZR     B4,DFM3     OC = IF( OC = OC.FM ) THEN OC.FA          CCG22         68     A
SQZ                S"OC"  OC.RFA                            ELSE OC.RFA         CCG22         69     A
SQZ                                                                             CCG22         70     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ       DFM3     BX7    X6                                                  CCG17        151     A
SQZ                LX7    R1.RKP-R1.RJP                                       SQZ          808     A
SQZ                IX6    X6+X7                                               SQZ          809     A
SQZ                S"OC"  OC.IA                                               SQZ          810         I
          -CCG18
SQZ                RJ     URI                                                 SQZ          811     A
SQZ                EQ     SRCH                                                SQZ          812     A
SQZ        FM      SPACE  3,7                                                 SQZ          813         I
          -CCG17
SQZ                PROCESS (FM,RFM)                                           SQZ          814         I
          -CCG17
SQZ                BX6    X1+X2                                               SQZ          815         I
          -CCG17
SQZ                LX6    59-RN.ZP                                            SQZ          816         I
          -CCG17
SQZ                MI     X6,ZERO     IF Z[RJ] ! Z[RK]                        SQZ          817         I
          -CCG17
SQZ                EQ     SRCH                                                SQZ          818         I
          -CCG17
SQZ                                                                           CCG17        152     A
SQZ       *        CHECK FLOATING MULTIPLY FOR POSSIBLE REDUCTION             CCG17        153     A
SQZ                                                                           CCG17        154     A
SQZ        FM1     SB4    "OC"-OC.DFM                                         CCG17        155     A
SQZ                EQ     B4,FM2      IF OC = DFM                             CCG17        156         I
          -CCG22
SQZ                AX6    48                                                  CCG17        157         I
          -CCG22
SQZ                NZ     B4,FM2      IF OC " OC.DFM                          CCG22         71     A
SQZ                LX7    13                                                  CCG22         72     A
SQZ                AX7    13                                                  CCG22         73     A
SQZ                ZR     X7,ZERO     IF DFM OF 2.**N  ( =0 )                 CCG22         74     A
SQZ                EQ     SRCH                                                CCG22         75     A
SQZ                                                                           CCG22         76     A
SQZ        FM2     BX6    X7                                                  CCG22         77     A
SQZ                AX6    48                                                  CCG22         78     A
SQZ                ZR     X6,ZERO     IF EXP[VAL] = 0                         CCG17        158     A
SQZ                                                                           CCG17        159     A
SQZ        FM2     SX6    2./1S45                                             CCG17        160         I
          -CCG22
SQZ                SX6    2./1S45                                             CCG22         79     A
SQZ                LX6    45                                                  CCG17        161     A
SQZ                IX3    X7-X6                                               CCG17        162     A
SQZ                NZ     X3,FM3      IF VAL " 2.0                            CCG17        163         I
          -CCG22
SQZ                                                                           CCG17        164         I
          -CCG22
SQZ                SA3    DFMA-OC.FM+"OC"                                     CCG17        165         I
          -CCG22
SQZ                BX6    -X0*X2                                              CCG17        166         I
          -CCG22
SQZ                LX6    R1.RJP                                              CCG17        167         I
          -CCG22

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                 S"OC"  X3+         OC = DFMA( OC )                          CCG17        168          I
          -CCG22
SQZ                 EQ     DFM2                                                  CCG17        169          I
          -CCG22    -CCG18
SQZ                 EQ     DFM3                                                  CCG18        151          I
          -CCG22
SQZ                                                                              CCG17        170          I
          -CCG22
SQZ        DFMA     CON    OC.FA,OC.RFA,OC.DFA                                   CCG17        171          I
          -CCG22
SQZ                 BX6    -X0*X2                                                CCG22         80     A
SQZ                 LX6    R1.RJP            */ SETUP X6 INCASE JUMP TO DFM4     CCG22         81     A
SQZ                 ZR     X3,DFM4     IF VAL = 2.0                              CCG22         82     A
SQZ                                                                              CCG22         83     A
SQZ                                                                              CCG17        172     A
SQZ        FM3      SX6    1./1S45                                               CCG17        173     A
SQZ                 LX6    45                                                    CCG17        174     A
SQZ                 IX3    X7-X6                                                 CCG17        175     A
SQZ                 IX7    X7+X6                                                 CCG17        176     A
SQZ                 SA1    "RN"+X2                                               CCG17        177     A
SQZ                 BX6   -X0*X2                                                 CCG17        178     A
SQZ                 LX6    R1.RJP                                                CCG17        179     A
SQZ                 ZR     X3,XMT0     IF VAL = 1.0                              CCG17        180     A
SQZ                 NZ     X7,SRCH     IF VAL " -1.0                             CCG17        181     A
SQZ                 EQ     DFM2                                                  CCG17        182     A
SQZ                                                                              SQZ          819     A
SQZ                 PROCESS (FD,RFD)                                             SQZ          820     A
SQZ                 LX1    59-RN.ZP                                              SQZ          821     A
SQZ                 MI     X1,ZERO     IF Z[RJ]                                  SQZ          822     A
SQZ                 LX1    RN.ZP-RN.PZP                                          SQZ          823     A
SQZ                 MI     X1,ZERO     IF PZ[RJ]                                 SQZ          824     A
SQZ                 EQ     SRCH                                                  SQZ          825          I
          -CCG17
SQZ                 BX7    X2                                                    CCG17        183     A
SQZ                 LX7    59-RN.CP                                              CCG17        184     A
SQZ                 PL     X7,SRCH     IF ^C[RK]                                 CCG17        185     A
SQZ                 LX2    -RN.TPP                                               CCG17        186          I
          -CCG18
SQZ                 SA3    X2                                                    CCG17        187          I
          -CCG18
SQZ                 LX7    1+RN.CP-RN.TPP                                        CCG18        152     A
SQZ                 SA3    X7                                                    CCG18        153     A
SQZ                 RJ     DCV                                                   CCG17        188     A
SQZ                 SA2    A1          (X2) = ^CONSTANT OPERAND                  CCG22         84     A
SQZ                 EQ     FM3         GO CHECK FOR X/1. , X/-1.                 CCG17        189     A
SQZ                                                                              SQZ          826     A
SQZ                 PROCESS OR                                                   SQZ          827     A
SQZ                 BX6    X1*X2                                                 SQZ          828          I
          -CCG17
SQZ                 LX6    59-RN.ZP                                              SQZ          829          I
          -CCG17
SQZ                 MI     X6,ZERO     IF Z[RJ] & Z[RK]                          SQZ          830          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
          -CCG17
SQZ                IX7    X1-X2                                          SQZ          831     A
SQZ                ZR     X7,C.XMT    IF RJ = RK                         SQZ          832     A
SQZ                BX6    X1+X2                                          SQZ          833     A
SQZ                LX6    59-RN.ZP                                       SQZ          834     A
SQZ                MI     X6,C.XMT    IF Z[RJ] ! Z[RK]                   SQZ          835     A
SQZ                LX6    RN.ZP-RN.XMTCP                                 SQZ          836     A
SQZ                PL     X6,SRCH                                        SQZ          837         I
          -CCG17
SQZ                PL     X6,SRCH     IF ^(XMTC[RJ] ! XMTC[RK])          CCG17        190     A
SQZ                S"OC"  OC.IMP                                         SQZ          838     A
SQZ                SB5    R1.RJP                                         SQZ          839     A
SQZ                EQ     CAO                                            SQZ          840     A
SQZ                                                                     SQZ          841     A
SQZ                PROCESS XOR                                           SQZ          842     A
SQZ                BX6    X1-X2                                          SQZ          843     A
SQZ                ZR     X6,ZERO     IF RND(RJ) = RND(RK)               SQZ          844     A
SQZ                BX6    X1*X2                                          SQZ          845         I
          -CCG17
SQZ                LX6    59-RN.ZP                                       SQZ          846         I
          -CCG17
SQZ                MI     X6,ZERO     IF Z[RJ] & Z[RK]                   SQZ          847         I
          -CCG17
SQZ                BX6    X1+X2                                          SQZ          848     A
SQZ                LX6    59-RN.ZP                                       SQZ          849     A
SQZ                MI     X6,C.XMT    IF Z[RJ] ! Z[RK]                   SQZ          850     A
SQZ                LX6    RN.ZP-RN.XMTCP                                 SQZ          851     A
SQZ                PL     X6,SRCH     IF ^XMTC[RJ] & ^XMTC[RK]           SQZ          852         I
          -CCG17
SQZ                PL     X6,SRCH     IF ^(XMTC[RJ] ! XMTC[RK])          CCG17        191     A
SQZ                S"OC"  OC.EQV                                         SQZ          853     A
SQZ                SB5    R1.RJP                                         SQZ          854     A
SQZ                EQ     CAO                                            SQZ          855     A
SQZ       IADD     SPACE  2                                              SQZ          856     A
SQZ       IA0      RJ     GCV         GET CONSTANT VALUES                SQZ          857         I
          -CCG17
SQZ                IX6    X6+X7       VAL[RI] = VAL[RJ] + VAL[RK]        SQZ          858         I
          -CCG17
SQZ                EQ     CCV                                            SQZ          859         I
          -CCG17
SQZ                                                                     SQZ          860         I
          -CCG17
SQZ                PROCESS IA                                            SQZ          861     A
SQZ                BX7    X1*X2                                          SQZ          862         I
          -CCG17
SQZ                LX7    59-RN.CP                                       SQZ          863         I
          -CCG17
SQZ                MI     X7,IA0      IF C[RJ] & C[RK]                   SQZ          864         I
          -CCG17
SQZ                BX6    X1+X2                                          SQZ          865     A
SQZ                LX6    59-RN.ZP                                       SQZ          866     A
SQZ                MI     X6,C.XMT    IF Z[RJ] ! Z[RK]                   SQZ          867     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                LX6    RN.ZP-RN.XMTCP                                          SQZ          868     A
SQZ                MI     X6,FA1      IF XMTC[RJ] ! XMTC[RK]                      SQZ          869     A
SQZ                LX6    RN.XMTCP-RN.CP                                          SQZ          870          I
          -FCC0514
SQZ                PL     X6,SRCH                                                 SQZ          871          I
          -FCC0514
SQZ                                                                               FCC0514      214     A
SQZ                BX7    X4                                                      FCC0716       12     A
SQZ                LX7    59-D.USIP                                               FCC0716       13     A
SQZ                MI     X7,IA1      IF USI[DI]    */ USI OR REDUCED IAZ/ISZ     FCC0716       14     A
SQZ                LX1    -RN.TPP                                                 FCC0514      215     A
SQZ                SA3    X1          R1J = TP[RJ]                                FCC0514      216     A
SQZ                SB5    OC.IS                                                   FCC0514      217     A
SQZ                UX7    B4,X3                                                   FCC0514      218     A
SQZ                LX1    RN.TPP                                                  FCC0514      219     A
SQZ                NE     B4,B5,IA1   IF OC[R1J] " OC.IS                          FCC0514      220     A
SQZ                                                                               FCC0514      221     A
SQZ                BX7    X5-X3                                                   FCC0514      222     A
SQZ                LX7    -R1.RKP                                                 FCC0514      223     A
SQZ                BX7    -X0*X7                                                  FCC0514      224     A
SQZ                LX3    -R1.RJP     RJX = RJ[R1J]                               FCC0514      225     A
SQZ                ZR     X7,I.XMT    IF RK[R1] = RK[R1J]  */ REDUCE (J-K)+K TO J FCC0514      226          I
          -FCC1660X
SQZ                NZ     X7,IA1      IF RK[R1] " RK[R1J]  */ NOT (J-K)+K         FCC1660X       6     A
SQZ                                                                               FCC1660X       7     A
SQZ                BX7    X1                                                      FCC1660X       8     A
SQZ                LX7    59-RN.INCP                                              FCC1660X       9     A
SQZ                PL     X7,I.XMT    IF ^INC[R1J]                                FCC1660X      10     A
SQZ                SX7    A3                                                      FCC1660X      11     A
SQZ                SB5    X3                                                      FCC1660X      12     A
SQZ                SA3    JPXA                                                    FCC1660X      13     A
SQZ                IX7    X7-X3                                                   FCC1660X      14     A
SQZ                SX3    B5                                                      FCC1660X      15     A
SQZ                PL     X7,I.XMT    IF R1J > JPXA */ IN THE SAME BLOCK          FCC1660X      16     A
SQZ                                                                               FCC0514      227     A
SQZ       IA1      LX6    RN.XMTCP-RN.CP                                          FCC0514      228     A
SQZ                PL     X6,SRCH     IF ^( C(RJ] ! C[RK] )                       FCC0514      229     A
SQZ                EQ     CIA                                                     SQZ          872     A
SQZ       IAZ      SPACE  3,14                                                    SQZ          873     A
SQZ      *         IAZ - ADD ZERO TO OPERAND PRIOR TO USE IN A CONDITIONAL JUMP   SQZ          874     A
SQZ                                                                               SQZ          875     A
SQZ                PROCESS IAZ                                                    SQZ          876     A
SQZ                BX7    X1*X2                                                   SQZ          877          I
          -CCG17
SQZ                LX7    59-RN.CP                                                SQZ          878          I
          -CCG17
SQZ                MI     X7,IA0      IF C[RJ] & C[RK]                            SQZ          879          I
          -CCG17
SQZ                MX7    2                                                       SQZ          880     A
SQZ                BX6    X1+X2       RJK = RJ ! RK                               SQZ          881     A
SQZ                BX3    X6                                                      CCG40         17     A
SQZ                LX3    59-RN.ZP                                                CCG40         18     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                PL     X3,SRCH     IF _Z[RJK]  */ IAZ INVENTED AT CIA4          CCG40        19     A
SQZ                LX6    58-RN.IACIP                                              SQZ         882     A
SQZ                BX7    X7*X6                                                    SQZ         883     A
SQZ                NZ     X7,C.XMT    IF IAC[RJK] ! INC[RJK]                       SQZ         884          I
          -FCC0716
SQZ                NZ     X7,IAZ1     IF IAC[RJK] ! INC[RJK]                       FCC0716      15     A
SQZ                LX6    1+RN.IACIP-RN.XMTCP                                      SQZ         885     A
SQZ                PL     X6,SRCH     IF ^XMTC[RJK]                                SQZ         886     A
SQZ                EQ     FA1                                                      SQZ         887     A
SQZ                                                                               FCC0716      16     A
SQZ       IAZ1     BX6    X2          RW = RKW                                     FCC0716      17     A
SQZ                LX6    59-RN.ZP                                                 FCC0716      18     A
SQZ                PL     X6,IAZ2     IF ^Z[RKW]                                   FCC0716      19     A
SQZ                BX6    X1          RW = RJW                                     FCC0716      20     A
SQZ                LX6    59-RN.ZP                                                 FCC0716      21     A
SQZ       IAZ2     LX6    1+RN.ZP-RN.TPP                                           FCC0716      22     A
SQZ                SA3    X6+B2       DP = TP[RW] + 2                              FCC0716      23     A
SQZ                SX7    B1                                                       FCC0716      24     A
SQZ                LX7    D.USIP                                                   FCC0716      25     A
SQZ                BX6    X7+X3       USI[DP] = 1   */ INHIBIT FURTHER REDUCTION   FCC0716      26     A
SQZ                SA6    A3                                                       FCC0716      27     A
SQZ                EQ     C.XMT                     */ GO CHANGE IAZ TO AN *XMT*   FCC0716      28     A
SQZ       DFA      SPACE  2,14                                                     SQZ         888     A
SQZ                PROCESS DFA                                                     SQZ         889     A
SQZ                BX6    X1+X2                                                    SQZ         890     A
SQZ                LX6    59-RN.ZP                                                 SQZ         891     A
SQZ                MI     X6,ZERO     IF Z[RJ] ! Z[RK]                             SQZ         892     A
SQZ                LX6    RN.ZP-RN.XMTCP                                           SQZ         893     A
SQZ                MI     X6,FA1      IF XMTC[RJ] ! XMTC[RK]                       SQZ         894     A
SQZ                EQ     SRCH                                                     SQZ         895     A
SQZ                                                                               SQZ         896     A
SQZ                PROCESS (FA,RFA)                                                SQZ         897     A
SQZ                BX6    X1*X2                                                    SQZ         898          I
          -CCG17
SQZ                LX6    59-RN.CP                                                 SQZ         899          I
          -CCG17
SQZ                MI     X6,ECO      IF C[RJ] & C[RK]                             SQZ         900          I
          -CCG17
SQZ                BX6    X1+X2                                                    SQZ         901     A
SQZ                LX6    59-RN.ZP                                                 SQZ         902     A
SQZ                MI     X6,F.XMT    IF Z[RJ] ! Z[RK]                             SQZ         903     A
SQZ                LX6    RN.ZP-RN.XMTCP                                           SQZ         904     A
SQZ                BX7    X1-X2                                                    SQZ         905     A
SQZ                MI     X6,FA1      IF XMTC[RJ] ! XMTC[RK]                       SQZ         906     A
SQZ                NZ     X7,SRCH     IF RJ " RK    */ NOT X+X                     SQZ         907     A
SQZ                SA3    "RN"+B6                                                  SQZ         908     A
SQZ                SX6    B1                                                       SQZ         909     A
SQZ                LX6    RN.NRP                                                   SQZ         910     A
SQZ                BX7    X6+X3       NR[RND(RI)] = 1  */ INDICATE NORMALIZE NOT   SQZ         911     A
SQZ                SA7    A3                                                       SQZ         912     A
SQZ                EQ     SRCH                                                     SQZ         913     A
SQZ                                                                               SQZ         914     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ       FA1      S"OC"  "OC"+1      OC = OC + 1   */ CHANGE ADD TO A SUBTRACT          SQZ          915     A
SQZ                SB5    R1.RKP                                                          SQZ          916     A
SQZ       CAO      SPACE  3,18                                                            SQZ          917     A
SQZ      **        CAO - CHANGE COMMUTATIVE-ADDITIVE OPERATION TO APPROPRIATE             SQZ          918     A
SQZ      *         SUBTRACTIVE OPERATION.                                                 SQZ          919     A
SQZ      *                                                                                SQZ          920     A
SQZ      *         ENTRY  (B3) = NEW OPCODE                                               SQZ          921     A
SQZ      *                (B5) = RJ/RK SHIFT COUNT                                        SQZ          922     A
SQZ      *                (X1/X2) = RND ENTRIES FOR RJ,RK                                 SQZ          923     A
SQZ      *                (X5) = R1 WORD                                                  SQZ          924     A
SQZ      *                (B6) = RI                                                       SQZ          925     A
SQZ                                                                                       SQZ          926     A
SQZ       CAO      BX6    X1                                                              SQZ          927     A
SQZ                LX5    -R1.RKP                                                         SQZ          928     A
SQZ                LX6    59-RN.XMTCP                                                     SQZ          929     A
SQZ                MI     X6,CAO1     IF XMTC[RJ]                                         SQZ          930     A
SQZ                BX1    X2                                                              SQZ          931     A
SQZ                LX5    R1.RKP-R1.RJP                                                   SQZ          932     A
SQZ                                                                                       SQZ          933     A
SQZ       CAO1     BX5    -X0*X5                                                          SQZ          934     A
SQZ                LX1    -RN.TPP                                                         SQZ          935     A
SQZ                SA2    X1          R1P = TP[RJ/K]                                      SQZ          936     A
SQZ                LX2    -R1.RJP                                                         SQZ          937     A
SQZ                BX1    -X0*X2      NEW R NUM                                           SQZ          938     A
SQZ                LX7    B5,X1                                                           SQZ          939     A
SQZ                SB5    B5-3*R1.RIL                                                     SQZ          940     A
SQZ                AX5    B5,X5                                                           SQZ          941     A
SQZ                BX6    X5+X7                                                           SQZ          942     A
SQZ                                                                                       SQZ          943     A
SQZ       CAO2     RJ     URI         UPDATE INSTRUCTIONS                                 SQZ          944         I
         -FCC0716
SQZ       CAO2     SX3    B1                                                              FCC0716       29     A
SQZ                LX3    D.USIP                                                          FCC0716       30     A
SQZ                BX7    X3*X4                                                           FCC0716       31     A
SQZ                LX7    -D.USIP                                                         FCC0716       32     A
SQZ                SB4    X7          U = USI[DI]                                         FCC0716       33     A
SQZ                RJ     URI         UPDATE INSTRUCTION                                  FCC0716       34     A
SQZ                SX7    B4                                                              FCC0716       35     A
SQZ                LX7    D.USIP                                                          FCC0716       36     A
SQZ                BX7    X7+X4       USI[DI] = USI[DI] ! U                               FCC0716       37     A
SQZ                SA7    A4                                                              FCC0716       38     A
SQZ                BX4    X7                                                              FCC0716       39     A
SQZ                LX6    -R1.RJP                                                         SQZ          945     A
SQZ                BX7    -X0*X6                                                          SQZ          946     A
SQZ                SA1    "RN"+X7     (X1) = RND(RJ)                                      SQZ          947     A
SQZ                LX6    R1.RJP-R1.RKP                                                   SQZ          948     A
SQZ                BX7    -X0*X6                                                          SQZ          949     A
SQZ                SA2    "RN"+X7     (X2) = RND(RK)                                      SQZ          950     A
SQZ                LX4    59-D.CMP                                                        SQZ          951     A
SQZ                PL     X4,CAO3     IF ^CM[DI]                                          SQZ          952     A
SQZ                SA4    A5-B2       DI = R1 - 2                                         SQZ          953     A
SQZ                EQ     MSL1        */ ADJUST RJ,RK BEFORE PROCESSING                   SQZ          954     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                                                                                  SQZ          955     A
SQZ       CAO3     LX4     1+D.CMP                                                   SQZ          956     A
SQZ                JP      SBB.JT+"OC" JUMP( OC )    */ PROCESS NEW INSTRUCTION      SQZ          957     A
SQZ       I.XMT    SPACE   3                                                         FCC0514      230     A
SQZ      **        I.XMT - CHANGE IA/IS COMBINATION TO AN *XMT/XMTC*                 FCC0514      231     A
SQZ      *                                                                          FCC0514      232     A
SQZ      *         ENTRY   (X3) = RJX , *RJ* OF *XMT/XMTC*                           FCC0514      233     A
SQZ                                                                                  FCC0514      234     A
SQZ       I.XMTC   S"OC"   OC.XMTC     OC = OC.XMTC                                  FCC0514      235     A
SQZ                EQ      I.XMT1                                                    FCC0514      236     A
SQZ                                                                                  FCC0514      237     A
SQZ       I.XMT    S"OC"   OC.XMT      OC = OC.XMT                                   FCC0514      238     A
SQZ       I.XMT1   BX6     -X0*X3                                                    FCC0514      239     A
SQZ                SA1     "RN"+X6     (X1) = RND(RJX)                               FCC0514      240     A
SQZ                LX6     R1.RJP                                                    FCC0514      241     A
SQZ                RJ      URI         URI(  OC  RI,RJX )                            FCC0514      242     A
SQZ                JP      SBB.JT+"OC" JUMP(SBB.JT(OC))                              FCC0514      243     A
SQZ       STR      TITLE   MASTER SQUEEZE LOOP - TYPE I SUBTRACTIVE INSTRUCTIONS     SQZ          958     A
SQZ                PROCESS STR         NAND  ( ^RJ & RK )                            SQZ          959     A
SQZ                BX6     X1-X2                                                     SQZ          960     A
SQZ                LX2     59-RN.ZP                                                  SQZ          961     A
SQZ                MI      X2,ZERO     IF Z[RK]                                      SQZ          962     A
SQZ                ZR      X6,ZERO     IF RJ = RK                                    SQZ          963     A
SQZ                LX1     59-RN.ZP                                                  SQZ          964     A
SQZ                MI      X1,C.XMT    IF Z[RJ]                                      SQZ          965     A
SQZ                S"OC"   OC.AND                                                    SQZ          966     A
SQZ                EQ      EQV1                                                      SQZ          967     A
SQZ       IMP      SPACE   3                                                         SQZ          968     A
SQZ                PROCESS IMP         NOR  ( ^RJ ! RK )                             SQZ          969     A
SQZ                BX6     X1-X2                                                     SQZ          970     A
SQZ                ZR      X6,FMA60    IF RJ = RK                                    SQZ          971     A
SQZ                LX1     59-RN.ZP                                                  SQZ          972     A
SQZ                MI      X1,FMA60    IF Z[RJ]                                      SQZ          973     A
SQZ                LX2     59-RN.ZP                                                  SQZ          974     A
SQZ                MI      X2,C.XMTC   IF Z[RK]                                      SQZ          975     A
SQZ                S"OC"   OC.OR                                                     SQZ          976     A
SQZ                EQ      EQV1                                                      SQZ          977     A
SQZ       EQV      SPACE   3                                                         SQZ          978     A
SQZ                PROCESS EQV         NXOR  ( ^RJ XOR RK )                          SQZ          979     A
SQZ                BX6     X1*X2                                                     SQZ          980         I
         -CCG17
SQZ                LX6     59-RN.ZP                                                  SQZ          981         I
         -CCG17
SQZ                IX7     X1-X2                                                     SQZ          982     A
SQZ                MI      X6,FMA60    IF Z[RJ] & Z[RK]                              SQZ          983         I
         -CCG17
SQZ                ZR      X7,FMA60    IF RJ = RK                                    SQZ          984     A
SQZ                BX6     X1+X2                                                     SQZ          985     A
SQZ                LX6     59-RN.ZP                                                  SQZ          986     A
SQZ                MI      X6,C.XMTC   IF Z[RJ] ! Z[RK]                              SQZ          987     A
SQZ                LX1     59-RN.ZP                                                  SQZ          988     A
SQZ                LX2     59-RN.ZP                                                  SQZ          989     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                 S"OC"  OC.XOR                                          SQZ          990     A
SQZ                                                                        SQZ          991     A
SQZ       *         CHANGE TO OPERATION NOT INVOLVING THE COMPLEMENT       SQZ          992     A
SQZ                                                                        SQZ          993     A
SQZ        EQV1     LX1    RN.ZP-RN.XMTCP                                  SQZ          994     A
SQZ                 PL     X1,SRCH     IF ^XMTC[RJ]                        SQZ          995     A
SQZ                 LX1    1+RN.ZP-RN.TPP                                  SQZ          996         I
           -FCC0716
SQZ                 LX1    1+RN.XMTCP-RN.TPP                               FCC0716       40     A
SQZ                 SA3    X1          R1P = TP[RJ]                        SQZ          997     A
SQZ                 LX5    -R1.RKP                                         SQZ          998     A
SQZ                 BX5    -X0*X5      RK                                  SQZ          999     A
SQZ                 LX3    -R1.RJP                                         SQZ         1000     A
SQZ                 BX6    -X0*X3      RJ = RJ[R1P]                        SQZ         1001     A
SQZ                 EQ     SUB1                                            SQZ         1002     A
SQZ        SUBTRACT SPACE  3,8                                             SQZ         1003     A
SQZ                 PROCESS DFS                                            SQZ         1004     A
SQZ                 BX6    X1+X2                                           SQZ         1005     A
SQZ                 LX6    59-RN.ZP                                        SQZ         1006     A
SQZ                 IX7    X1-X2                                           SQZ         1007     A
SQZ                 MI     X6,ZERO     IF Z[RJ] ! Z[RK]                    SQZ         1008     A
SQZ                 NZ     X7,SUB      IF RJ " RK                          SQZ         1009     A
SQZ                 EQ     ZERO                                            SQZ         1010     A
SQZ                                                                        SQZ         1011     A
SQZ                                                                        CCG17        192     A
SQZ                 PROCESS (FS,RFS)                                       SQZ         1012     A
SQZ                 IX6    X1-X2                                           SQZ         1013     A
SQZ                 ZR     X6,ZERO     IF RJ = RK                          SQZ         1014     A
SQZ                 BX7    X1*X2                                           SQZ         1015         I
           -CCG17
SQZ                 LX7    59-RN.CP                                        SQZ         1016         I
           -CCG17
SQZ                 MI     X7,ECO      IF C[RJ] & C[RK]                    SQZ         1017         I
           -CCG17
SQZ                 LX2    59-RN.ZP                                        SQZ         1018     A
SQZ                 MI     X2,F.XMT    IF Z[RK]                            SQZ         1019     A
SQZ                 LX1    59-RN.ZP                                        SQZ         1020     A
SQZ                 MI     X1,F.XMTC   IF Z[RJ]                            SQZ         1021     A
SQZ                 LX2    1+RN.ZP                                         SQZ         1022     A
SQZ                 EQ     SUB                                             SQZ         1023     A
SQZ                                                                        SQZ         1024     A
SQZ        IS0      RJ     GCV         GET CONSTANT VALUES                 SQZ         1025         I
           -CCG17
SQZ                 IX6    X6-X7       VAL[RI] = VAL[RJ] - VAL[RK]         SQZ         1026         I
           -CCG17
SQZ                 EQ     CCV                                             SQZ         1027         I
           -CCG17
SQZ                                                                        SQZ         1028         I
           -CCG17
SQZ                 PROCESS IS                                             SQZ         1029     A
SQZ                 BX6    X1-X2                                           SQZ         1030     A
SQZ                 SB4    X6                                              SQZ         1031     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                ZR     B4,ZERO     IF RN[RJ] = RN[RK]                          SQZ          1032     A
SQZ                BX7    X1*X2                                                   SQZ          1033          I
          -CCG17
SQZ                LX7    59-RN.CP                                                SQZ          1034          I
          -CCG17
SQZ                MI     X7,IS0      IF C[RJ] & C[RK]                            SQZ          1035          I
          -CCG17
SQZ                LX2    59-RN.ZP                                                SQZ          1036     A
SQZ                MI     X2,C.XMT    IF Z[RK]                                    SQZ          1037     A
SQZ                LX1    59-RN.ZP                                                SQZ          1038     A
SQZ                MI     X1,C.XMTC   IF Z[RJ]                                    SQZ          1039     A
SQZ                LX1    1+RN.ZP                                                 SQZ          1040     A
SQZ                LX2    1+RN.ZP                                                 SQZ          1041     A
SQZ                BX6    X1+X2                                                   SQZ          1042          I
          -FCC0514
SQZ                                                                               FCC0514       244     A
SQZ       *        CHECK FOR AND REDUCE COMBINATIONS OF THE FORM (J+K)-K , ETC.  FCC0514       245     A
SQZ                                                                               FCC0514       246     A
SQZ                BX7    X4                                                      FCC0716        41     A
SQZ                LX7    59-D.USIP                                               FCC0716        42     A
SQZ                MI     X7,IS4      IF USI[DI]    */ USI OR REDUCED IAZ/ISZ    FCC0716        43     A
SQZ                LX1    -RN.TPP                                                 FCC0514       247     A
SQZ                SA3    X1          R1J = TP[RJ]  */ CHECK J OPERAND            FCC0514       248     A
SQZ                SB5    OC.IA                                                   FCC0514       249     A
SQZ                UX7    B4,X3                                                   FCC0514       250     A
SQZ                LX1    RN.TPP                                                  FCC0514       251     A
SQZ                NE     B4,B5,IS1   IF OC[R1J] " OC.IA                          FCC0514       252     A
SQZ                                                                               FCC0514       253     A
SQZ                LX3    -R1.RJP                                                 FCC0514       254     A
SQZ                BX7    X2-X3                                                   FCC0514       255     A
SQZ                LX3    R1.RJP-R1.RKP     RJX = RK[R1J]                         FCC0514       256     A
SQZ                BX7    -X0*X7                                                  FCC0514       257     A
SQZ                ZR     X7,I.XMT    IF RJ[R1J] = RK  */ REDUCE (J+K)-J TO K     FCC0514       258          I
          -FCC1660X
SQZ                ZR     X7,IS1A     IF RJ[R1J] = RK  */ REDUCE (J+K)-J TO K     FCC1660X       17     A
SQZ                BX7    X2-X3                                                   FCC0514       259     A
SQZ                BX7    -X0*X7                                                  FCC0514       260     A
SQZ                NZ     X7,IS4      IF RK[R1J] " RK */ NOT (K+J)-K              FCC0514       261     A
SQZ                LX3    R1.RKP-R1.RJP     RJX = RJ[R1J]                         FCC0514       262     A
SQZ                EQ     I.XMT                                                   FCC0514       263          I
          -FCC1660X
SQZ                                                                               FCC1660X       18     A
SQZ        IS1A    BX6    X1          RJW = RND(RJ)                               FCC1660X       19     A
SQZ                EQ     IS3A                                                    FCC1660X       20     A
SQZ                                                                               FCC0514       264     A
SQZ        IS1     NE     B4,B3,IS2   IF OC[R1J] " OC.IS                          FCC0514       265     A
SQZ                LX3    -R1.RJP                                                 FCC0514       266     A
SQZ                BX7    X2-X3                                                   FCC0514       267     A
SQZ                BX7    -X0*X7                                                  FCC0514       268     A
SQZ                NZ     X7,IS4      IF RJ[R1J] " RK  */ NOT (J-K)-J             FCC0514       269     A
SQZ                LX3    R1.RJP-R1.RKP     RJX = RK[R1P]                         FCC0514       270     A
SQZ                EQ     I.XMTC                                                  FCC0514       271     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                                                                                  FCC0514      272     A
SQZ       IS2      LX2    -RN.TPP                                                   FCC0514      273     A
SQZ                SA3    X2          R1K = TP[RK]  */ CHECK K OPERAND              FCC0514      274     A
SQZ                LX2    RN.TPP                                                    FCC0514      275     A
SQZ                UX7    B4,X3                                                     FCC0514      276     A
SQZ                NE     B4,B5,IS3   IF OC[R1K] " OC.IA                           FCC0514      277     A
SQZ                LX3    -R1.RJP                                                   FCC0514      278     A
SQZ                BX7    X1-X3                                                     FCC0514      279     A
SQZ                LX3    R1.RJP-R1.RKP     RJX = RK[R1K]                           FCC0514      280     A
SQZ                BX7    -X0*X7                                                    FCC0514      281     A
SQZ                ZR     X7,I.XMTC   IF RJ[R1K] = RJ  */ REDUCE J-(J+K) TO -K     FCC0514      282     A
SQZ                BX7    X1-X3                                                     FCC0514      283     A
SQZ                BX7    -X0*X7                                                    FCC0514      284     A
SQZ                NZ     X7,IS4      IF RK[R1K] " RJ  */ NOT J-(K+J)              FCC0514      285     A
SQZ                LX3    R1.RKP-R1.RJP     RJX = RJ[R1K]                           FCC0514      286     A
SQZ                EQ     I.XMTC                                                    FCC0514      287     A
SQZ                                                                                  FCC0514      288     A
SQZ       IS3      NE     B4,B3,IS4   IF OC[R1K] " OC.IS                           FCC0514      289     A
SQZ                BX7    X3-X5                                                     FCC0514      290     A
SQZ                LX7    -R1.RJP                                                   FCC0514      291     A
SQZ                BX7    -X0*X7                                                    FCC0514      292     A
SQZ                NZ     X7,IS4      IF RJ[R1K] " RJ */ NOT J-(J-K)               FCC0514      293     A
SQZ                LX3    -R1.RKP     RJX = RK[R1K]                                FCC0514      294     A
SQZ                BX6    X2                                                        FCC1660X      21     A
SQZ                                                                                  FCC1660X      22     A
SQZ       IS3A     LX6    59-RN.INCP                                                FCC1660X      23     A
SQZ                PL     X6,I.XMT    IF ^INC[RJW]                                 FCC1660X      24     A
SQZ                SX7    A3                                                        FCC1660X      25     A
SQZ                SB5    X3                                                        FCC1660X      26     A
SQZ                SA3    JPXA                                                      FCC1660X      27     A
SQZ                IX6    X7-X3                                                     FCC1660X      28     A
SQZ                SX3    B5                                                        FCC1660X      29     A
SQZ                PL     X6,I.XMT    IF R1P > JPXA */ IN THE SAME BLOCK           FCC1660X      30     A
SQZ                EQ     IS4                                                       FCC1660X      31     A
SQZ                EQ     I.XMT                                                     FCC0514      295     A
SQZ                                                                                  FCC0514      296     A
SQZ        IS4     BX6    X1+X2                                                     FCC0514      297           I
           -FCC1719X
SQZ                LX6    59-RN.CP                                                  SQZ         1043           I
           -FCC1719X
SQZ                MI     X6,CIA      IF C[RJ] ! C[RK]                             SQZ         1044           I
           -FCC1719X
SQZ                LX6    1+RN.CP+58-RN.IACIP                                       SQZ         1045           I
           -FCC1424  -FCC1719X
SQZ                MX7    2                                                         SQZ         1046           I
           -FCC1424  -FCC1719X
SQZ                BX7    X7*X6                                                     SQZ         1047           I
           -FCC1424  -FCC1719X
SQZ                ZR     X7,SUB      IF ^( IACI[RJ] ! IACI[RK] )                  SQZ         1048           I
           -FCC1424  -FCC1719X
SQZ                LX6    RN.CP-RN.INCP                                             FCC1424        4           I
           -FCC1719X

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ       IS4     BX6    X2                                                  FCC1719X      5     A
SQZ               LX6    59-RN.CP                                            FCC1719X      6     A
SQZ               MI     X6,CIA      IF C[RK]                                FCC1719X      7     A
SQZ               BX6    X1+X2                                               FCC1719X      8     A
SQZ               LX6    59-RN.INCP                                          FCC1719X      9     A
SQZ               PL     X6,SUB      IF ^( INC[RJ] ! INC[RK] )               FCC1424       5     A
SQZ                                                                          SQZ        1049     A
SQZ      *        I = I + 1  $  IF( I .RO. N ) , MARK *IS* AS *INC*          SQZ        1050     A
SQZ                                                                          SQZ        1051     A
SQZ               SA3    "RN"+B6                                             SQZ        1052     A
SQZ               SX7    B1                                                  SQZ        1053     A
SQZ               LX7    RN.INCP                                             SQZ        1054     A
SQZ               BX6    X7+X3       INC[RI] = 1                             SQZ        1055     A
SQZ               SA6    A3                                                  SQZ        1056     A
SQZ                                                                          SQZ        1057     A
SQZ       SUB     LX2    59-RN.XMTCP                                         SQZ        1058     A
SQZ               PL     X2,SRCH     IF ^XMTC[RK]                            SQZ        1059     A
SQZ               LX2    1+RN.XMTCP-RN.TPP                                   SQZ        1060     A
SQZ               SA3    X2          R1P = TP[RK]                            SQZ        1061     A
SQZ               S"OC"  "OC"-B1     CHANGE SUBTRACT TO AN ADD               SQZ        1062     A
SQZ               LX5    -R1.RJP                                             SQZ        1063     A
SQZ               BX6    -X0*X5      RJ                                      SQZ        1064     A
SQZ               LX3    -R1.RJP                                             SQZ        1065     A
SQZ               BX5    -X0*X3      RK = RJ[R1P]                            SQZ        1066     A
SQZ                                                                          SQZ        1067     A
SQZ       SUB1    IX7    X6-X5                                               SQZ        1068     A
SQZ               PL     X7,SUB2     IF RJ \ RK                              SQZ        1069     A
SQZ               SX7    X5                                                  SQZ        1070     A
SQZ               BX5    X6                                                  SQZ        1071     A
SQZ               LX6    X7                                                  SQZ        1072     A
SQZ                                                                          SQZ        1073     A
SQZ       SUB2    LX6    R1.RJP                                              SQZ        1074     A
SQZ               LX5    R1.RKP                                              SQZ        1075     A
SQZ               BX6    X5+X6                                               SQZ        1076     A
SQZ               EQ     CAO2        GO UPDATE AND PROCESS CHANGED INSTRUCTION SQZ        1077     A
SQZ       ISZ     SPACE  2                                                   SQZ        1078          I
         -CCG17
SQZ       ISZ     SPACE  2,14                                                CCG17       193     A
SQZ      *        ISZ - ADD 0 AND CHANGE SIGN PRIOR TO USE IN CONDITIONAL JUMP SQZ        1079     A
SQZ      *        THE RJ OPERAND IS ASSUMED TO BE A *CLR  RJ,0* IN THE BELOW. SQZ        1080          I
         -CCG14
SQZ      *        THE RJ OPERAND SHOULD CLR RJ,0.                            CCG14         7          I
         -CCG17
SQZ      *        THE RJ OPERAND SHOULD BE A  *CLR  RJ,0*                    CCG17       194     A
SQZ                                                                          SQZ        1081     A
SQZ               PROCESS ISZ                                                SQZ        1082     A
SQZ               BX6    X1*X2                                               SQZ        1083          I
         -CCG17
SQZ               LX6    59-RN.CP                                            SQZ        1084          I
         -CCG17
SQZ               MI     X6,IS0      IF C[RJ] & C[RK]                        SQZ        1085          I
         -CCG17

         0        1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                                                                                  SQZ        1086     A
SQZ     *         SEE IF THE PRED ( *RK* ) IS AN *IS*, AND IF SO ELIMINATE THE       SQZ        1087     A
SQZ     *         *ISZ* BY REVERSING THE OPERANDS.                                   SQZ        1088     A
SQZ                                                                                  SQZ        1089     A
SQZ               LX2    -RN.TPP                                                     SQZ        1090     A
SQZ               SA3    X2          R1P = TP[RK]                                    SQZ        1091     A
SQZ               S"OC"  OC.IS                                                       SQZ        1092     A
SQZ               UX6    B4,X3                                                       SQZ        1093     A
SQZ               NE     B4,"OC",SRCH      IF OC[R1P] " OC.IS                        SQZ        1094     A
SQZ               LX1    59-RN.ZP                                                    CCG14         8     A
SQZ               PL     X1,SRCH     IF ^ Z[RJ]                                      CCG14         9     A
SQZ               LX6    -R1.RJP                                                     SQZ        1095     A
SQZ               BX7    -X0*X6      J = RJ[R1P]                                     SQZ        1096     A
SQZ               SA1    "RN"+X7     RJW = RND(J)                                    SQZ        1097     A
SQZ               LX2    59-RN.IACP+RN.TPP                                           SQZ        1098     A
SQZ               MI     X2,ISZ1     IF IAC[RK]                                      SQZ        1099     A
SQZ               MX2    2                                                           SQZ        1100     A
SQZ               LX1    58-RN.IACIP                                                 SQZ        1101     A
SQZ               BX2    X2*X1                                                       SQZ        1102     A
SQZ               ZR     X2,SRCH     IF ^( IAC[RJW] ! INC[RJW] )                     SQZ        1103     A
SQZ                                                                                  SQZ        1104     A
SQZ       ISZ1    LX3    -R1.RKP                                                     SQZ        1105     A
SQZ               BX3    -X0*X3                                                      SQZ        1106     A
SQZ               LX7    R1.RKP                                                      SQZ        1107     A
SQZ               LX3    R1.RJP                                                      SQZ        1108     A
SQZ               BX6    X3+X7                                                       SQZ        1109     A
SQZ               RJ     URI         URI(  IS  RI,RK[R1P],RJ[R1P] )                  SQZ        1110     A
SQZ               SX3    B1                                                          FCC0716      44     A
SQZ               LX3    D.USIP                                                      FCC0716      45     A
SQZ               BX6    X3+X4       USI[DI] = 1   */ INHIBIT FURTHER REDUCTION      FCC0716      46     A
SQZ               SA6    A4                                                          FCC0716      47     A
SQZ               BX4    X3+X4                                                       FCC0716      48     A
SQZ               EQ     SRCH                                                        SQZ        1111     A
SQZ       CIA     EJECT                                                              SQZ        1112     A
SQZ      **       CIA - CHECK IA/IS INSTRUCTION WITH A CONSTANT OPERAND              SQZ        1113     A
SQZ      *        SET *IAC* BIT IF NOT AN INCREMENT ( I = I +-CON )                  SQZ        1114     A
SQZ                                                                                  SQZ        1115     A
SQZ       RN.CP   EQU    58                                                          SQZ        1116     A
SQZ       CIA     BX7    X2          RCW = RK;  RVW = RJ                             SQZ        1117          I
         -FCC0715
SQZ       CIA     BX7    X4                                                          FCC0715       6     A
SQZ               LX7    59-D.INCP                                                   FCC0715       7     A
SQZ               MI     X7,CIA2A    IF INC[DI]                                      FCC0715       8     A
SQZ                                                                                  FCC0715       9     A
SQZ               BX7    X2          RCW = RK;  RVW = RJ                             FCC0715      10     A
SQZ               LX3    B1,X1                                                       SQZ        1118     A
SQZ               PL     X3,CIA1     IF ^C[RJ]                                       SQZ        1119     A
SQZ               BX7    X1          RCW = RJ                                        SQZ        1120     A
SQZ               BX1    X2          RVW = RK                                        SQZ        1121     A
SQZ                                                                                  SQZ        1122     A
SQZ       CIA1    LX7    -RN.TPP                                                     SQZ        1123     A
SQZ               SA3    X7          R1P = TP[RCW]                                   SQZ        1124     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                UX6    B4,X3                                                    SQZ         1125     A
SQZ                LX3    -R1.INP                                                  SQZ         1126     A
SQZ                SB4    B4-OC.FMA                                                SQZ         1127     A
SQZ                SX6    X3-60                                                    SQZ         1128     A
SQZ                NZ     B4,CIA2     IF OC[R1P] " OC.FMA                          SQZ         1129     A
SQZ                PL     X6,CIA4     IF IN[R1P] \ 60  */ A -0                     SQZ         1130     A
SQZ                                                                                SQZ         1131     A
SQZ       *        CONSTANT OPERAND NOT -0, AND CANT BE A *CLR*, SET IAC/INC BIT   SQZ         1132     A
SQZ                                                                                SQZ         1133     A
SQZ        CIA2    SA3    A4+B2       R1S = DI + 2  */ NEXT INSTRUCTION            SQZ         1134     A
SQZ                SB4    RN.IACP     BIT = IAC                                    SQZ         1135     A
SQZ                BX6    X5-X3                                                    SQZ         1136     A
SQZ                BX7    -X0*X6                                                   SQZ         1137          I
          -FCC1025
SQZ                NZ     X7,CIA3     IF RI[R1S] " RI[R1]                          SQZ         1138          I
          -FCC1025
SQZ                BX6    -X0*X6                                                   FCC1025       14     A
SQZ                NZ     X6,CIA3     IF RI[R1S] " RI[R1]                          FCC1025       15     A
SQZ                UX6    B3,X3                                                    SQZ         1139     A
SQZ                SX7    B3-OC.ST                                                 SQZ         1140          I
          -FCC1025
SQZ                NZ     X7,CIA3     IF OC[R1S] " OC.ST                           SQZ         1141          I
          -FCC1025
SQZ                SB5    OC.ST                                                    FCC1025       16     A
SQZ                NE     B3,B5,CIA3  IF OC[R1S] " OC.ST  */  NOT ST RI,         FCC1025       17     A
SQZ                                                                                SQZ         1142     A
SQZ                SA3    A3+B1       R2S = R1S + 1                                SQZ         1143     A
SQZ                LX1    -RN.TPP                                                  SQZ         1144     A
SQZ                SA2    X1+B1       R2P = TP[RVW] + 1                            SQZ         1145     A
SQZ                BX6    X2-X3                                                    SQZ         1146     A
SQZ                NZ     X6,CIA3     IF [R2P] " [R2S]  */ NOT I = I + CON         SQZ         1147     A
SQZ                SA2    A2-B1       R1P = R2P - 1                                FCC1025       18     A
SQZ                UX6    B3,X2                                                    FCC1025       19     A
SQZ                SB5    OC.SA                                                    FCC1025       20     A
SQZ                EQ     B3,B5,CIA5  IF OC[R1P] = OC.SA                           FCC1025       21     A
SQZ                SB4    RN.INCP     BIT = INC                                    SQZ         1148          I
          -FCC0715
SQZ                                                                                FCC0715       11     A
SQZ        CIA2A   SB4    RN.INCP     BIT = INC                                    FCC0715       12     A
SQZ                                                                                SQZ         1149     A
SQZ        CIA3    SA3    "RN"+B6                                                  SQZ         1150     A
SQZ                SX6    B1                                                       SQZ         1151     A
SQZ                LX7    B4,X6                                                    SQZ         1152     A
SQZ                BX6    X7+X3       BIT[RND(RI)] = 1                             SQZ         1153     A
SQZ                SA6    A3                                                       SQZ         1154     A
SQZ                EQ     SRCH                                                     SQZ         1155     A
SQZ                                                                                SQZ         1156     A
SQZ       *        CON IS -0, CHANGE TO IAZ/ISZ SO *GPO* DOESNT THINK ITS AN *INC  SQZ         1157     A
SQZ                                                                                SQZ         1158     A
SQZ        CIA4    BX6    X0*X5                                                    SQZ         1159     A
SQZ                S"OC"  "OC"-OC.IA+OC.IAZ                                        SQZ         1160     A
SQZ                RJ     URI         CHANGE INSTRUCTION                           SQZ         1161     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                EQ     SRCH                                                    SQZ         1162     A
SQZ                                                                               FCC1025       22     A
SQZ      *         DEF/SA/INC/ST VAR SEQUENCE, CHANGE THE *INC* TO A *STT*.       FCC1025       23     A
SQZ                                                                               FCC1025       24     A
SQZ       CIA5     SA3    X7          R1C = TP[RCW]                               FCC1025       25     A
SQZ                RJ     DCV         C = DCV( R1C )                              FCC1025       26     A
SQZ                BX6    X4                                                      FCC1025       27     A
SQZ                LX6    59-D.CMP                                                FCC1025       28     A
SQZ                AX6    59                                                      FCC1025       29     A
SQZ                BX7    -X6-X7      C = XOR( C , ^CM[DI] )                      FCC1025       30     A
SQZ                MX6    -IH.CAL                                                 FCC1025A       1     A
SQZ                BX7    -X6*X7                                                  FCC1025A       2     A
SQZ                LX2    -R1.RJP                                                 FCC1025       31     A
SQZ                BX6    -X0*X2      RF = RJ[R1P]                                FCC1025       32     A
SQZ                LX7    IH.CAP                                                  FCC1025       33     A
SQZ                LX6    IH.RFP                                                  FCC1025       34     A
SQZ                BX7    X6+X7                                                   FCC1025       35     A
SQZ                SA7    A5+B1       [R1+1] = IHW(RF,C,0)                        FCC1025       36     A
SQZ                S"OC"  OC.STT                                                  FCC1025       37     A
SQZ                MX6    0                                                       FCC1025       38     A
SQZ                RJ     URI         URI( STT  RI,RF,C )                         FCC1025       39     A
SQZ                EQ     .STT                                                    FCC1025       40     A
SQZ       PK/UP    TITLE  MASTER SQUEEZE LOOP - PACK/UNPACK                       SQZ         1163     A
SQZ                PROCESS PK                                                     SQZ         1164     A
SQZ                BX6    X1*X2                                                   SQZ         1165          I
          -FCC1085
SQZ                LX6    59-RN.ZP                                                SQZ         1166          I
          -FCC1085
SQZ                PL     X6,SRCH     IF ^ ( Z[RK] & Z[RJ] )                      SQZ         1167          I
          -FCC1085
SQZ                LX1    59-RN.ZP                                                FCC1085       14     A
SQZ                PL     X1,IRCH     IF ^Z[RJ]     */ DONT OVERSQZ RJ USES       FCC1085       15     A
SQZ                LX0    R1.RJP                                                  FCC1085       16     A
SQZ                BX6    X0*X5                                                   FCC1085       17     A
SQZ                BX5    X0*X5       RJ[R1] = 0    */ INCASE RJ " 0              FCC1085       18     A
SQZ                SA6    A5                                                      FCC1085       19     A
SQZ                LX0    -R1.RJP                                                 FCC1085       20     A
SQZ                LX2    59-RN.ZP                                                FCC1085       21     A
SQZ                PL     X2,SRCH     IF ^Z[RK]     */ NOT A PACKED  ZERO         FCC1085       22     A
SQZ                                                                               FCC1085       23     A
SQZ                SX7    B1                                                      SQZ         1168     A
SQZ                SA3    "RN"+B6                                                 SQZ         1169     A
SQZ                LX7    RN.PZP                                                  SQZ         1170     A
SQZ                BX6    X7+X3                                                   SQZ         1171     A
SQZ                SA6    A3          PZ[RI] = 1                                  SQZ         1172     A
SQZ                EQ     SRCH                                                    SQZ         1173     A
SQZ       NR       SPACE  3                                                       SQZ         1174     A
SQZ                PROCESS (NR,RNZ)                                               SQZ         1175     A
SQZ                BX3    X2                                                      CCG17        195     A
SQZ                LX3    59-RN.CP                                                CCG17        196     A
SQZ                MI     X3,ECO4     IF C[RND(RK)]                               CCG17        197          I
          -CCG18

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ              LX1    59-RN.ZP                                          SQZ        1176         I
          -CCG18
SQZ              PL     X3,NR0      IF ^C[RND(RK)]                        CCG18       154     A
SQZ              SB4    X1                                                CCG18       155     A
SQZ              LE     B4,B1,ECO.NR      IF RJ @ 1                       CCG18       156     A
SQZ              EQ     ECO4                                              CCG18       157     A
SQZ                                                                       CCG18       158     A
SQZ       NR0    LX1    59-RN.ZP                                          CCG18       159     A
SQZ              PL     X1,NR1      IF ^Z[RJ]                             SQZ        1177     A
SQZ              BX3    X2                                                SQZ        1178         I
          -CCG17
SQZ              LX3    59-RN.ZP                                          SQZ        1179         I
          -CCG17
SQZ              MI     X3,ZERO     IF Z[RK]                              SQZ        1180         I
          -CCG17
SQZ              LX3    RN.ZP-RN.PZP                                      SQZ        1181         I
          -CCG17
SQZ              LX3    RN.CP-RN.PZP                                      CCG17       198     A
SQZ              MI     X3,ZERO     IF PZ[RK]                             SQZ        1182     A
SQZ              LX3    RN.PZP-RN.NRP                                     SQZ        1183     A
SQZ              SX6    B1                                                SQZ        1184     A
SQZ              MI     X3,SRCH4    IF NR[RK]                             SQZ        1185     A
SQZ              SA3    "RN"+B6                                           SQZ        1186     A
SQZ              LX6    RN.NRP                                            SQZ        1187     A
SQZ              BX7    X6+X3       NR[RI] = 1                            SQZ        1188     A
SQZ              SA7    A3                                                SQZ        1189     A
SQZ              EQ     SRCH                                              SQZ        1190     A
SQZ                                                                       SQZ        1191     A
SQZ       NR1    SB4    X1                                                SQZ        1192     A
SQZ              NE     B4,B1,UP4   IF RK " 1     */ NOT *PNR*            SQZ        1193         I
          -FCC0514
SQZ              NE     B4,B1,UP4   IF RJ " 1     */ NOT *PNR*            FCC0514     298     A
SQZ                                                                       FCC0514     299     A
SQZ              LX2    -RN.TPP                                           SQZ        1194     A
SQZ              SA2    X2          R1P = TP[RK]                          SQZ        1195     A
SQZ              UX7    B4,X2                                             SQZ        1196     A
SQZ              NE     B4,"OC",SRCH      IF OC[R1P] " OC[R1]             SQZ        1197     A
SQZ              LX7    -R1.RJP                                           SQZ        1198     A
SQZ              BX6    -X0*X7                                            SQZ        1199     A
SQZ              SX3    B1                                                SQZ        1200     A
SQZ              SB4    X6                                                SQZ        1201     A
SQZ              GT     B4,B1,UP4   IF RJ[R1P] > 1                        SQZ        1202     A
SQZ                                                                       SQZ        1203     A
SQZ              LX3    R1.RJP                                            SQZ        1204     A
SQZ              BX6    X2+X3       RJ[NRP] = 1                           SQZ        1205         I
          -FCC0514
SQZ              BX6    X2+X3       RJ[R1P] = 1                           FCC0514     300     A
SQZ              SA6    A2                                                SQZ        1206     A
SQZ              BX2    X2+X3                                             SQZ        1207     A
SQZ              EQ     SRCH4             */ GO ELIMINITE THE INSTRUCTION SQZ        1208     A
SQZ       UP     SPACE  3                                                 SQZ        1209     A
SQZ              PROCESS UP                                               SQZ        1210     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                LX2    59-RN.ZP                                                  SQZ        1211        I
          -CCG17
SQZ                MI     X2,ZERO     IF Z[RK]                                      SQZ        1212        I
          -CCG17
SQZ                LX2    1+RN.ZP-RN.TPP                                            SQZ        1213        I
          -CCG17
SQZ                BX3    X2                                                        CCG17       199    A
SQZ                LX3    59-RN.CP                                                  CCG17       200    A
SQZ                MI     X3,ECO4     IF C[RK]                                      CCG17       201    A
SQZ                LX2    -RN.TPP                                                   CCG17       202    A
SQZ                SB4    B2+B2                                                     SQZ        1214    A
SQZ                SA3    X2+B4       R1S = TP[RK] + 4                              SQZ        1215    A
SQZ                SB5    A5                                                        SQZ        1216    A
SQZ                SA2    UPA                                                       SQZ        1217    A
SQZ                BX5    X2*X5       KEY = TYI(UP,0,RK,0)                          SQZ        1218    A
SQZ                                                                                SQZ        1219    A
SQZ       UP1      BX6    X2*X3                                                     SQZ        1220    A
SQZ                SA3    A3+B4       R1S = R1S + 4                                 SQZ        1221    A
SQZ                IX7    X5-X6                                                     SQZ        1222    A
SQZ                NZ     X7,UP1      IF [R1S-4] " KEY                              SQZ        1223    A
SQZ                SA5    A5                                                        SQZ        1224    A
SQZ                SB3    A3-B4                                                     SQZ        1225    A
SQZ                GE     B3,B5,UP4   IF R1S \ R1   */ INSTRUCTION NOT REDUNDANT    SQZ        1226    A
SQZ                                                                                SQZ        1227    A
SQZ                SA3    A3-B4                                                     SQZ        1228    A
SQZ                LX1    59-RN.ZP                                                  SQZ        1229    A
SQZ                MI     X1,UP2      IF Z[RJ]      */ BI,B0 CASE                   SQZ        1230    A
SQZ                LX3    -R1.RJP                                                   SQZ        1231    A
SQZ                BX2    -X0*X3                                                    SQZ        1232    A
SQZ                ZR     X2,UP4      IF Z[RJ[R1S]]  */ CANNOT HANDLE B0,BI CASE    SQZ        1233    A
SQZ                                                                                SQZ        1234    A
SQZ                MX6    60                                                        SQZ        1235    A
SQZ                SA6    A4+4        [DI+4] = -0   */ MARK FOLLOWING *RS* AS DEA   SQZ        1236    A
SQZ                SA2    "RN"+X2                                                   SQZ        1237    A
SQZ                BX6    X2                                                        SQZ        1238    A
SQZ                LX5    -R1.RJP                                                   SQZ        1239    A
SQZ                BX2    -X0*X5                                                    SQZ        1240    A
SQZ                SA6    "RN"+X2     RND(RJ[R1]) = RND(RJ[R1S])                    SQZ        1241    A
SQZ                LX3    R1.RJP                                                    SQZ        1242    A
SQZ                                                                                SQZ        1243    A
SQZ       UP2      BX2    -X0*X3                                                    SQZ        1244    A
SQZ                SA1    "RN"+X2                                                   SQZ        1245    A
SQZ                BX6    X1          RND(RI[R1]) = RND(RI[R1S])                    SQZ        1246    A
SQZ                SA3    A3+B2       DJ = R1S + 2                                  SQZ        1247    A
SQZ                SA6    "RN"+B6                                                   SQZ        1248    A
SQZ                BX6    X3+X4       DJ = DJ ! DI                                  SQZ        1249    A
SQZ                SA7    A4          [DI] = 0                                      SQZ        1250    A
SQZ                SA6    A3                                                        SQZ        1251    A
SQZ                EQ     MSL1                                                      SQZ        1252    A
SQZ                                                                                SQZ        1253    A
SQZ       UP4      LX5    R1.RJP-R1.RKP                                             SQZ        1254    A
SQZ                EQ     AUC.RJ                                                    SQZ        1255    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                                                                                  SQZ        1256    A
SQZ       UPA      VFD    12/-0,16/0,16/-0,16/0                                      SQZ        1257    A
SQZ       ILS      SPACE  3                                                          SQZ        1258    A
SQZ       TEMP     IFNE   0,0                                                        SQZ        1259    A
SQZ                PROCESS (ILS,IRS)                                                 SQZ        1260    A
SQZ                BX6    X1+X2                                                      SQZ        1261    A
SQZ                SB3    B3-OC.ILS                                                  SQZ        1262    A
SQZ                LX6    59-RN.CP                                                   SQZ        1263    A
SQZ                PL     X6,SRCH     IF ^( C[RJ] ! C[RK] )                          SQZ        1264    A
SQZ                BX7    X1*X2                                                      SQZ        1265        I
          -CCG17
SQZ                LX7    59-RN.CP                                                   SQZ        1266        I
          -CCG17
SQZ                PL     X7,ILS1     IF ^( C[RJ] & C[RK] )                          SQZ        1267        I
          -CCG17
SQZ                                                                                  SQZ        1268        I
          -CCG17
SQZ                RJ     GCV         GET CONSTANT VALUES                            SQZ        1269        I
          -CCG17
SQZ                SB4    X6                                                         SQZ        1270        I
          -CCG17
SQZ                LX6    B4,X7                                                      SQZ        1271        I
          -CCG17
SQZ                ZR     B3,CCV      VAL[RI] = SHIFT( VAL[RJ] , VAL[RK] )           SQZ        1272        I
          -CCG17
SQZ                AX6    B4,X7                                                      SQZ        1273        I
          -CCG17
SQZ                EQ     CCV                                                        SQZ        1274        I
          -CCG17
SQZ                                                                                  SQZ        1275        I
          -CCG17
SQZ       ILS1     LX2    59-RN.ZP                                                   SQZ        1276        I
          -CCG17
SQZ                LX2    59-RN.ZP                                                   CCG17       203    A
SQZ                MI     X2,ZERO     IF Z[RK]                                       SQZ        1277    A
SQZ                LX1    59-RN.ZP                                                   SQZ        1278    A
SQZ                PL     X1,SRCH     IF ^Z[RJ]                                      SQZ        1279    A
SQZ                BX1    X2                                                         SQZ        1280    A
SQZ                S"OC"  OC.XMT                                                     SQZ        1281    A
SQZ                EQ     C.XMT2      GO OUTPUT XMT( RI , RK )                       SQZ        1282    A
SQZ       TEMP     ELSE                                                              SQZ        1283    A
SQZ       IRCH     SPACE  4,8                                                        SQZ        1284    A
SQZ      *         THIS TEMPORARY CODE IS NECESSARY UNTIL MCG/JAM IS ABLE TO         SQZ        1285    A
SQZ      *         CODE SEQUENCES CONTAINING B-REGISTER JAMS.  THIS ILS/IRS          SQZ        1286    A
SQZ      *         PROCESSOR USES A SPECIAL SEARCH/SQUEEZE LOOP (IRCH) THAT          SQZ        1287    A
SQZ      *         SUPPRESSES ELIMINATION OF B-REG TRANSMITS (SHORT ADDS) PRE-       SQZ        1288    A
SQZ      *         CEDING INDEXED SHIFTS.  ONLY SA OR STT PREDECESSORS ARE           SQZ        1289        I
          -FCC0872
SQZ      *         CEDING INDEXED SHIFTS.  ONLY S, SA, OR STT PREDECESSORS ARE       FCC0872       4    A
SQZ      *         CORRECTLY HANDLED AND THEY MUST DIRECTLY PRECEDE THE SHIFT IN     SQZ        1290    A
SQZ      *         THE SEQUENCE ALONG WITH THE RS.                                   SQZ        1291    A
SQZ                                                                                  SQZ        1292    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                 PROCESS (ILS,IRS)                                       SQZ         1293     A
SQZ                 BX6    X1+X2                                            SQZ         1294         I
          -CCG17
SQZ                 SB3    B3-OC.ILS                                        SQZ         1295         I
          -CCG17
SQZ                 LX6    59-RN.CP                                         SQZ         1296         I
          -CCG17
SQZ                 PL     X6,IRCH     IF ^( C[RJ] ! C[RK] )                SQZ         1297         I
          -CCG17
SQZ                 BX7    X1*X2                                            SQZ         1298         I
          -CCG17
SQZ                 LX7    59-RN.CP                                         SQZ         1299         I
          -CCG17
SQZ                 PL     X7,ILS1     IF ^( C[RJ] & C[RK] )                SQZ         1300         I
          -CCG17
SQZ                                                                         SQZ         1301         I
          -CCG17
SQZ                 RJ     GCV         GET CONSTANT VALUES                  SQZ         1302         I
          -CCG17
SQZ                 SB4    X6                                               SQZ         1303         I
          -CCG17
SQZ                 LX6    B4,X7                                            SQZ         1304         I
          -CCG17
SQZ                 ZR     B3,ICV      VAL[RI] = SHIFT( VAL[RJ] , VAL[RK] ) SQZ         1305         I
          -CCG17
SQZ                 AX6    B4,X7                                            SQZ         1306         I
          -CCG17
SQZ                 EQ     ICV                                              SQZ         1307         I
          -CCG17
SQZ                                                                         SQZ         1308         I
          -CCG17
SQZ        ILS1     LX2    59-RN.ZP                                         SQZ         1309         I
          -CCG17
SQZ                 LX2    59-RN.ZP                                         CCG17        204     A
SQZ                 MI     X2,ZERO     IF Z[RK]                             SQZ         1310     A
SQZ                 LX1    59-RN.ZP                                         SQZ         1311     A
SQZ                 PL     X1,IRCH     IF ^Z[RJ]                            SQZ         1312     A
SQZ                 BX1    X2                                               SQZ         1313     A
SQZ                 S"OC"  OC.XMT                                           SQZ         1314     A
SQZ                 EQ     C.XMT2      GO OUTPUT XMT( RI , RK )             SQZ         1315     A
SQZ                 SPACE  4,8                                              SQZ         1316     A
SQZ       **        ICV - TEMPORARY CCV FOR ILS/IRS                         SQZ         1317         I
          -CCG17
SQZ       *         FAIL EXIT IS IRCH RATHER THAN SRCH.                     SQZ         1318         I
          -CCG17
SQZ       *         ENTRY  (X6) = VAL , TO BE REFORMED AS *S*/*FMA* INSTRUCTION SQZ     1319         I
          -CCG17
SQZ                                                                         SQZ         1320         I
          -CCG17
SQZ       *         TRY TO REDUCE CONSTANT TO A *FMA*                       SQZ         1321         I
          -CCG17
SQZ                                                                         SQZ         1322         I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
          -CCG17
SQZ        ICV      PL     X6,ICV1     IF VAL > 0                                  SQZ        1323        I
          -CCG17
SQZ                 CX7    X6                                                      SQZ        1324        I
          -CCG17
SQZ                 MX3    1                                                       SQZ        1325        I
          -CCG17
SQZ                 SB4    X7-1                                                    SQZ        1326        I
          -CCG17
SQZ                 AX7    B4,X3                                                   SQZ        1327        I
          -CCG17
SQZ                 BX3    X7-X6                                                   SQZ        1328        I
          -CCG17
SQZ                 NZ     X3,ICV1     IF VAL " MASK( COUNT(VAL) )                 SQZ        1329        I
          -CCG17
SQZ                 S"OC"  OC.FMA                                                  SQZ        1330        I
          -CCG17
SQZ                 SX6    B4+B1                                                   SQZ        1331        I
          -CCG17
SQZ                 EQ     ICV2                                                    SQZ        1332        I
          -CCG17
SQZ                                                                                SQZ        1333        I
          -CCG17
SQZ       *         TRY TO REDUCE CONSTANT TO A *S*                                SQZ        1334        I
          -CCG17
SQZ                                                                                SQZ        1335        I
          -CCG17
SQZ        ICV1     ZR     X6,ZERO     IF VAL = 0                                  SQZ        1336        I
          -CCG17
SQZ                 MX3    42                                                      SQZ        1337        I
          -CCG17
SQZ                 LX7    X6                                                      SQZ        1338        I
          -CCG17
SQZ                 BX6    -X3*X6                                                  SQZ        1339        I
          -CCG17
SQZ                 AX7    17                                                      SQZ        1340        I
          -CCG17
SQZ                 NZ     X7,IRCH     IF SHIFT(VAL,-17) " 0                       SQZ        1341        I
          -CCG17
SQZ                 S"OC"  OC.S                                                    SQZ        1342        I
          -CCG17
SQZ                                                                                SQZ        1343        I
          -CCG17
SQZ        ICV2     LX6    R1.INP                                                  SQZ        1344        I
          -CCG17
SQZ                 RJ     URI                                                     SQZ        1345        I
          -CCG17
SQZ                 MX7    0                                                       SQZ        1346        I
          -CCG17
SQZ                 SA7    A5+B1       [R1+1] = 0                                  SQZ        1347        I
          -CCG17
SQZ                 EQ     FMA                                                     SQZ        1348        I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

                -CCG17
SQZ                     SPACE  4,8                                                  SQZ        1349          I
                -CCG17
SQZ              IRCH   BX6    X0*X5       KEY = TYI(OC,RJ,RK,0)                    SQZ        1350     A
SQZ                     LX5    -R1.RKP                                              SQZ        1351     A
SQZ                     BX3    -X0*X5                                               SQZ        1352     A
SQZ                     SB4    B2+B2       (B4) = 4                                 SQZ        1353     A
SQZ                     LX5    R1.RKP-R1.RJP                                        SQZ        1354     A
SQZ                     SA2    "RN"+X3                                              SQZ        1355     A
SQZ                     BX3    -X0*X5                                               SQZ        1356     A
SQZ                     LX2    -RN.TPP                                              SQZ        1357     A
SQZ                     SA1    "RN"+X3                                              SQZ        1358     A
SQZ                     SB5    X2                                                   SQZ        1359     A
SQZ                     LX1    -RN.TPP                                              SQZ        1360     A
SQZ                     SB3    X1          R1S = TP[RJ]                             SQZ        1361     A
SQZ                     GE     B3,B5,IRCH1 IF R1S \ TP[RK]                          SQZ        1362     A
SQZ                     SB3    B5          R1S = TP[RK]                             SQZ        1363     A
SQZ                                                                                 SQZ        1364     A
SQZ              IRCH1  SA3    B3+4        R1S = R1S + 4                            SQZ        1365     A
SQZ                     SB5    A2                                                   SQZ        1366     A
SQZ                                                                                 SQZ        1367     A
SQZ             *       SEARCH FORWARD FROM MAX OF DEF ADDRESS OF OPERANDS.         SQZ        1368     A
SQZ                                                                                 SQZ        1369     A
SQZ              IRCH2  BX7    X0*X3                                                SQZ        1370     A
SQZ                     SA3    A3+B4                                                SQZ        1371     A
SQZ                     IX5    X6-X7                                                SQZ        1372     A
SQZ                     NZ     X5,IRCH2    IF [R1S] " KEY                           SQZ        1373     A
SQZ                     SA2    A3-B2                                                SQZ        1374     A
SQZ                     SB3    A3-B4                                                SQZ        1375     A
SQZ                     ZR     X2,IRCH2    IF [DS] = 0   */ DEAD INSTRUCTION        SQZ        1376     A
SQZ                     SB3    A5-B3                                                SQZ        1377     A
SQZ                     GT     B3,SRCH3    IF R1S < R1M  */ INSTRUCTION IS REDUNDANT SQZ        1378     A
SQZ                                                                                 SQZ        1379     A
SQZ                     SB3    A5-8        R1J = R1 - 8                             SQZ        1380          I
                -CCG16
SQZ                     SA2    A5          R1J = R1      */ SEARCH FOR J OPERAND    CCG16       245     A
SQZ              IRCH2A SA2    A2-B4       R1J = R1J - 4                            CCG16       246     A
SQZ                     UX7    B3,X2                                                CCG16       247     A
SQZ                     SX2    B3-OC.RS                                             CCG16       248     A
SQZ                     NZ     X2,IRCH2A   IF OC[R1J] " OC.RS                       CCG16       249     A
SQZ                     SB3    A2-B4       R1J = R1J - 4 */ POINT TO J OPERAND      CCG16       250     A
SQZ                     MX2    -RN.USESL                                            SQZ        1381     A
SQZ                     LX1    RN.TPP-RN.USESP                                      SQZ        1382     A
SQZ                     BX5    -X2*X1                                               SQZ        1383     A
SQZ                     ZR     X5,IRCH5    IF USES[RND(RJ)] = 0                     SQZ        1384     A
SQZ                     SA2    B3+B2       DI = R1J + 2 ;  DIW = [DI]               SQZ        1385     A
SQZ                     SA3    B3          R1I = R1J ;  R1W = [R1I]                 SQZ        1386     A
SQZ                     MX7    1                                                    SQZ        1387     A
SQZ                     NZ     X2,IRCH5    IF DIW .NE. 0 */ INS. NOT SQUEEZED       SQZ        1388     A
SQZ                     LX7    1+D.PRSP                                             SQZ        1389     A
SQZ                     UX5    B6,X3       OCI = OC[R1W]                            SQZ        1390     A
SQZ                     SA3    =XF.RDT+B6                                           SQZ        1391     A

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                BX6    X3+X7                                                  SQZ        1392    A
SQZ                SA6    A2          [DI] = RDT(OCI) .OR. D.PRS                 SQZ        1393    A
SQZ                LX6    58-D.TYP                                               SQZ        1394    A
SQZ                MI     X6,IRCH3    IF TY[DIW] = 3  */ STT                     SQZ        1395    A
SQZ                LX2    B1,X6                                                  FCC0872       5    A
SQZ                MI     X2,IRCH4    IF TY[DIW] = 2  */ S                       FCC0872       6    A
SQZ                BX6    X5                                                     SQZ        1396    A
SQZ                LX6    -R1.RJP                                                SQZ        1397    A
SQZ                BX2    -X0*X6                                                 SQZ        1398    A
SQZ                SA3    "RN"+X2                                                SQZ        1399    A
SQZ                MX6    1                                                      SQZ        1400    A
SQZ                LX6    1+RN.USESP                                             SQZ        1401    A
SQZ                IX6    X3+X6                                                  SQZ        1402    A
SQZ                SA6    A3          USES[RND(RJ[R1W])] = USES[RND(RJ[R1W])] + 1 SQZ        1403    A
SQZ                EQ     IRCH4                                                  SQZ        1404    A
SQZ                                                                              SQZ        1405    A
SQZ       IRCH3    SA3    B3+B1       R2I = R1I + 1 ;  R2W = [R2]                SQZ        1406    A
SQZ                LX3    -IH.RFP                                                SQZ        1407    A
SQZ                SA3    "RN"+X3                                                SQZ        1408    A
SQZ                MX6    1                                                      SQZ        1409    A
SQZ                LX6    1+RN.USESP                                             SQZ        1410    A
SQZ                IX6    X3+X6                                                  SQZ        1411    A
SQZ                SA6    A3          USES[RND(RF[R2W])] = USES[RND(RF[R2W])] + 1 SQZ        1412    A
SQZ                                                                              SQZ        1413    A
SQZ       IRCH4    SA3    =XF.RDT+OC.RS                                          SQZ        1414    A
SQZ                BX6    X3                                                     SQZ        1415    A
SQZ                SA6    A2+B4       [DI+4] = RDT(OC.RS)                        SQZ        1416    A
SQZ                BX5    -X0*X5      RIJ = RI[R1W]                              SQZ        1417    A
SQZ                SA2    A5                                                     SQZ        1418    A
SQZ                LX2    -R1.RJP                                                SQZ        1419    A
SQZ                BX2    X0*X2                                                  SQZ        1420    A
SQZ                BX6    X2+X5                                                  SQZ        1421    A
SQZ                LX6    R1.RJP                                                 SQZ        1422    A
SQZ                SA6    A2                                                     SQZ        1423    A
SQZ                SX2    X5                                                     SQZ        1424    A
SQZ                LX5    RN.RNP                                                 SQZ        1425    A
SQZ                LX7    RN.USESP-D.PRSP                                        SQZ        1426    A
SQZ                BX6    X5+X7                                                  SQZ        1427    A
SQZ                SX5    B3                                                     SQZ        1428    A
SQZ                LX5    RN.TPP                                                 SQZ        1429    A
SQZ                BX6    X6+X5                                                  SQZ        1430    A
SQZ                SA6    "RN"+X2     RND(RJ) = RNW(RIJ,1,R1J)                   SQZ        1431    A
SQZ                SA2    B5                                                     SQZ        1432    A
SQZ                IX7    X7+X2       USES[RK] = USES[RK] +1                     SQZ        1433    A
SQZ                SA7    A2                                                     SQZ        1434    A
SQZ                EQ     MSL1                                                   SQZ        1435    A
SQZ                                                                              SQZ        1436    A
SQZ       IRCH5    LX1    RN.USESP                                               SQZ        1437    A
SQZ                SX5    B1                                                     SQZ        1438    A
SQZ                LX5    RN.USESP                                               SQZ        1439    A
SQZ                IX6    X5+X1       USES[RJ] = USES[RJ] + 1                    SQZ        1440    A
SQZ                SA6    A1                                                     SQZ        1441    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ              SA2    B5                                                    SQZ        1442    A
SQZ              IX7    X5+X2       USES[RK] = USES[RK] + 1                   SQZ        1443    A
SQZ              SA7    A2                                                    SQZ        1444    A
SQZ              EQ     MSL1                                                  SQZ        1445    A
SQZ       TEMP   ENDIF                                                        SQZ        1446    A
SQZ       SRCH   TITLE  SRCH - SEARCH FOR PREVIOUS OCCURANCE OF TYPE I INST   SQZ        1447    A
SQZ      **      SRCH - SEARCH FOR PREVIOUS OCCURRENCE OF TYPE I INSTRUCTION. SQZ        1448    A
SQZ      *       ENTRY  (X5) = R1 WORD , (B6) = RI                           SQZ        1449    A
SQZ                                                                          SQZ        1450    A
SQZ              PROCESS (CX,SS)                                             SQZ        1451    A
SQZ       SRCH   BX6    X0*X5       KEY = TYI(OC,RJ,RK,0)  */ SEARCH KEY    SQZ        1452    A
SQZ              LX5    -R1.RKP                                             SQZ        1453    A
SQZ              BX3    -X0*X5                                              SQZ        1454    A
SQZ              SB4    B2+B2       (B4) = 4                                SQZ        1455    A
SQZ              LX5    R1.RKP-R1.RJP                                       SQZ        1456    A
SQZ              SA2    "RN"+X3                                             SQZ        1457    A
SQZ              BX3    -X0*X5                                              SQZ        1458    A
SQZ              LX2    -RN.TPP                                             SQZ        1459    A
SQZ              SA1    "RN"+X3                                             SQZ        1460    A
SQZ              SB5    X2                                                  SQZ        1461    A
SQZ              LX1    -RN.TPP                                             SQZ        1462    A
SQZ              SB3    X1                                                  SQZ        1463    A
SQZ              GE     B3,B5,SRCH1 R1S = MAX( TP[RJ] , TP[RK] ) */ MIN SRCH AD SQZ    1464    A
SQZ              SB3    B5+                                                 SQZ        1465    A
SQZ       SRCH1  SA3    B3+4        R1S = R1S + 4                           SQZ        1466    A
SQZ              SB5    A2                                                  SQZ        1467    A
SQZ                                                                          SQZ        1468    A
SQZ      *       SEARCH FORWARD FROM MAX OF DEF ADDRESS OF OPERANDS.         SQZ        1469    A
SQZ                                                                          SQZ        1470    A
SQZ       SRCH2  BX7    X0*X3                                               SQZ        1471    A
SQZ              SA3    A3+B4                                               SQZ        1472    A
SQZ              IX5    X6-X7                                               SQZ        1473    A
SQZ              NZ     X5,SRCH2    IF [R1S] " KEY                          SQZ        1474    A
SQZ              SA2    A3-B2                                               SQZ        1475    A
SQZ              SB3    A3-B4                                               SQZ        1476    A
SQZ              ZR     X2,SRCH2    IF [DS] = 0   */ DEAD INSTRUCTION       SQZ        1477    A
SQZ              SB3    A5-B3                                               SQZ        1478    A
SQZ              GT     B3,SRCH3    IF R1S < R1M  */ INSTRUCTION IS REDUNDANT SQZ      1479    A
SQZ                                                                          SQZ        1480    A
SQZ      *       INSTRUCTION NOT REDUNDANT, BUMP USES OF OPERANDS            SQZ        1481    A
SQZ                                                                          SQZ        1482    A
SQZ              LX1    RN.TPP                                              SQZ        1483    A
SQZ              SX5    B1                                                  SQZ        1484    A
SQZ              LX5    RN.USESP                                            SQZ        1485    A
SQZ              IX6    X5+X1       USES[RJ] = USES[RJ] + 1                 SQZ        1486    A
SQZ              SA6    A1                                                  SQZ        1487    A
SQZ              SA2    B5                                                  SQZ        1488    A
SQZ              IX7    X5+X2       USES[RK] = USES[RK] + 1                 SQZ        1489    A
SQZ              SA7    A2                                                  SQZ        1490    A
SQZ              EQ     MSL1                                                SQZ        1491    A
SQZ                                                                          SQZ        1492    A
SQZ      *       SRCH4 - REDUNDANT INSTRUCTION FOUND, ATTEMPT TO ELIMINATE IT SQZ       1493    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

SQZ       *         ENTRY   (X2) = WORD WITH RJ, RIGHT JUSTIFIED                    SQZ        1494     A
SQZ       *                 (X4) = INSTRUCTION DESCRIPTOR                           SQZ        1495     A
SQZ       *                 (B6) = RI OF CURRENT INSTRUCTION                        SQZ        1496     A
SQZ                                                                                 SQZ        1497     A
SQZ        SRCH3    SA2     A3-4        RJ = RI[R1S]                                SQZ        1498          I
          -CCG17
SQZ                 SA4     A4+         REFETCH DESCR                               SQZ        1499          I
          -CCG17
SQZ        SRCH3    SA1     A3-B2       DIS = R1S - 2                               CCG17       205     A
SQZ                 SA5     A5          REFETCH R1 WORD                             CCG17       206     A
SQZ                 SA4     A4          REFETCH DESCR                               CCG17       207     A
SQZ                 SA2     A1-B2       RJ = RI[R1S-4]                              CCG17       208     A
SQZ                 BX6     X1-X4                                                   CCG17       209     A
SQZ                 LX6     59-D.USIP                                               CCG17       210     A
SQZ                 PL      X6,SRCH4    IF USI[DIS] = USI[DI]                       CCG17       211     A
SQZ                 UX6     "OC",X5                                                 CCG17       212     A
SQZ                 SX6     "OC"-OC.IM                                              CCG17       213     A
SQZ                 NZ      X6,SRCH4    IF OC[R1] " OC.IM                           CCG17       214     A
SQZ                                                                                 CCG17       215     A
SQZ                 SX6     B1                                                      CCG17       216     A
SQZ                 LX6     D.USIP                                                  CCG17       217     A
SQZ                 BX7     -X6*X1      USI[DIS] = 0  */ SAFE IF ONE IS ALWAYS EXEC CCG17       218     A
SQZ                 SA7     A1                                                      CCG17       219     A
SQZ                                                                                 SQZ        1500     A
SQZ                                                                                 SQZ        1501     A
SQZ        SRCH4    MX6     -D.STRSL                                                SQZ        1502     A
SQZ                 BX2     -X0*X2                                                  SQZ        1503     A
SQZ                 LX6     D.STRSP                                                 SQZ        1504     A
SQZ                 SA1     "RN"+X2                                                 SQZ        1505     A
SQZ                 BX7     -X6*X4                                                  SQZ        1506     A
SQZ                 NZ      X7,SRCH5    IF STRS[DI] " 0                             SQZ        1507     A
SQZ                 SA7     A4          [DI] = 0                                    SQZ        1508     A
SQZ                 BX6     X1                                                      SQZ        1509     A
SQZ                 SA6     "RN"+B6     RND(RI) = RND(RJ)                           SQZ        1510     A
SQZ                 EQ      MSL1                                                    SQZ        1511     A
SQZ                                                                                 SQZ        1512     A
SQZ       *         STRS[DI] " 0, CHANGE TO AN XMT AND LET THE *XMT* PROCESSOR      SQZ        1513     A
SQZ       *         DO THE REST OF THE DIRTY WORK.                                  SQZ        1514     A
SQZ                                                                                 SQZ        1515     A
SQZ        SRCH5    MX7     -RN.XMTP                                                SQZ        1516          I
          -FCC1413
SQZ        SRCH5    SA3     "RN"+B6                                                 FCC1413       5     A
SQZ                 LX3     59-RN.INCP                                              FCC1413       6     A
SQZ                 PL      X3,SRCH5A   IF ^INC[RI]                                 FCC1413       7     A
SQZ                 SX7     B1                                                      FCC1413       8     A
SQZ                 LX7     RN.INCP                                                 FCC1413       9     A
SQZ                 BX1     X7+X1       INC[RJ] = 1                                 FCC1413      10     A
SQZ                 LX7     RN.IACP-RN.INCP                                         FCC1413      11     A
SQZ                 BX6     -X7*X1      IAC[RJ] = 0                                 FCC1413      12     A
SQZ                 SA6     A1                                                      FCC1413      13     A
SQZ                                                                                 FCC1413      14     A
SQZ        SRCH5A   MX7     -RN.XMTP                                                FCC1413      15     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                BX1    -X7*X1      XMT[RJ] = 0   */ SO WE CAN SQUEEZE XMITS          SQZ        1517    A
SQZ                                                                                     SQZ        1518    A
SQZ       SRCH6    LX2    R1.RJP                                                        SQZ        1519    A
SQZ                S"OC"  OC.XMT                                                        SQZ        1520    A
SQZ                BX6    X2                                                            SQZ        1521    A
SQZ                RJ     URI         GENERATE AN XMT                                   SQZ        1522    A
SQZ                RJ     CTI         TRY AND ELIMINATE IT                              SQZ        1523    A
SQZ                EQ     AUC.RJ                                                        SQZ        1524    A
SQZ       C.XMT    SPACE  3                                                             SQZ        1525    A
SQZ      **        F.XMTC/F.XMT - FLOATING POINT ADD/SUB REDUCES TO AN XMT/XMTC         SQZ        1526    A
SQZ                                                                                     SQZ        1527    A
SQZ       F.XMTC   S"OC"  OC.XMTC                                                       SQZ        1528    A
SQZ                EQ     F.XMT1                                                        SQZ        1529    A
SQZ                                                                                     SQZ        1530    A
SQZ       F.XMT    S"OC"  OC.XMT                                                        SQZ        1531    A
SQZ       F.XMT1   SA3    "RN"+B6                                                       SQZ        1532    A
SQZ                SX6    B1                                                            SQZ        1533    A
SQZ                LX6    RN.NRP                                                        SQZ        1534    A
SQZ                BX7    X6+X3       NR[RI] = 1    */ RESULT IS NORMALIZED FLAG        SQZ        1535    A
SQZ                SA7    A3                                                            SQZ        1536    A
SQZ                EQ     C.XMT1                                                        SQZ        1537    A
SQZ       C.XMT    SPACE  3                                                             SQZ        1538    A
SQZ      *         C.XMT / C.XMTC - CHANGE INSTRUCTION TO XMT / XMTC                    SQZ        1539    A
SQZ                                                                                     SQZ        1540    A
SQZ       C.XMTC   S"OC"  OC.XMTC                                                       SQZ        1541    A
SQZ                EQ     C.XMT1                                                        SQZ        1542    A
SQZ                                                                                     SQZ        1543    A
SQZ       C.XMT    S"OC"  OC.XMT                                                        SQZ        1544    A
SQZ       C.XMT1   SA1    A1                                                            SQZ        1545    A
SQZ                LX1    59-RN.ZP                                                      SQZ        1546    A
SQZ                PL     X1,C.XMT2   IF ^Z[RJ]                                         SQZ        1547    A
SQZ                SA1    A2                                                            SQZ        1548    A
SQZ                LX1    59-RN.ZP                                                      SQZ        1549    A
SQZ       C.XMT2   LX1    1+RN.ZP                                                       SQZ        1550    A
SQZ                BX6    -X0*X1                                                        SQZ        1551    A
SQZ                LX6    R1.RJP                                                        SQZ        1552    A
SQZ                RJ     URI         UPDATE REDUCED INSTRUCTION                        SQZ        1553    A
SQZ                SA1    "RN"+X1     (X1) = RND(RJ)                                    SQZ        1554    A
SQZ                JP     SBB.JT+"OC" JUMP( OC )                                        SQZ        1555    A
SQZ       CTI      TITLE  CTI - CHECK TRANSMIT INSTRUCTION                              SQZ        1556    A
SQZ      **        CTI - CHECK *XMT* INSTRUCTION TO SEE IF IT CAN BE ELIMINATED.        SQZ        1557    A
SQZ      *                                                                             SQZ        1558    A
SQZ      *         ENTRY  (X1) = RND(RJ)                                                SQZ        1559    A
SQZ      *                X4,X5,B6 AS FROM THE MAIN LOOP                               SQZ        1560    A
SQZ      *                                                                             SQZ        1561    A
SQZ      *         EXIT   (X5) = [R1] IF XMT CANNOT BE ELIMINATED, ELSE TO MAIN L       SQZ        1562    A
SQZ                                                                                     SQZ        1563    A
SQZ       CTI      ROUTINE                                                              SQZ        1564    A
SQZ                                                                                     SQZ        1565    A
SQZ      *         ELIMINATE XMT/XMT SEQUENCES                                          SQZ        1566    A
SQZ                                                                                     SQZ        1567    A
SQZ       CTI0     LX1    59-RN.XMTP                                                    SQZ        1568    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                PL     X1,CTI1     IF ^XMT[RJ]                                  SQZ        1569     A
SQZ                LX1    1+RN.XMTP-RN.TPP                                         SQZ        1570     A
SQZ                SA2    X1          R1J = TP[RJ]                                 SQZ        1571     A
SQZ                S"OC"  OC.XMT                                                   SQZ        1572     A
SQZ                SX5    B6                                                       SQZ        1573     A
SQZ                LX2    -R1.RJP                                                  SQZ        1574     A
SQZ                BX3    -X0*X2      RJ = RJ[R1J]                                 SQZ        1575     A
SQZ                SA1    "RN"+X3                                                  SQZ        1576     A
SQZ                LX3    R1.RJP                                                   SQZ        1577     A
SQZ                PX6    "OC",X3                                                  SQZ        1578     A
SQZ                IX7    X6+X5                                                    SQZ        1579     A
SQZ                SA7    A5          [R1] = TYI(XMT,RJ,0,RI)                      SQZ        1580     A
SQZ                BX5    X6+X5                                                    SQZ        1581     A
SQZ                EQ     CTI0                                                     SQZ        1582     A
SQZ                                                                               SQZ        1583     A
SQZ       *        DETERMINE PRECEDENCE BIT SETTING AND JUMP TO PROCESSOR.         SQZ        1584     A
SQZ                                                                               SQZ        1585     A
SQZ        CTI1    LX1    1+RN.XMTP-RN.TPP                                         SQZ        1586     A
SQZ                SA3    X1+B2       DJ = [TP[RJ]+2]                              SQZ        1587     A
SQZ                MX2    -D.STRSL                                                 SQZ        1588     A
SQZ                LX4    -D.STRSP                                                 SQZ        1589     A
SQZ                BX6    -X2*X4                                                   SQZ        1590     A
SQZ                SB4    X6                                                       SQZ        1591     A
SQZ                LX3    -D.STRSP                                                 SQZ        1592     A
SQZ                BX7    -X2*X3                                                   SQZ        1593     A
SQZ                LX1    RN.TPP                                                   SQZ        1594     A
SQZ                SB5    X7                                                       SQZ        1595     A
SQZ                JP     CTI2+B4     JUMP( STRS[DI] )  */ 0 - 2 = -/PRS/PS        SQZ        1596     A
SQZ                                                                               SQZ        1597     A
SQZ        CTI2    EQ     CTI3        STRS[DI] = 0                                 SQZ        1598     A
SQZ                                                                               SQZ        1599     A
SQZ                ZR     B5,CTI4     IF STRS[DJ] = 0                              SQZ        1600     A
SQZ                EQ     CTI5                                                     SQZ        1601     A
SQZ                                                                               SQZ        1602     A
SQZ                ZR     B5,CTI6     IF PS[DI]                                    SQZ        1603     A
SQZ                EQ     B5,B1,CTI5  IF PS[DI] & PRS[DJ]                          SQZ        1604     A
SQZ                                                                               SQZ        1605     A
SQZ       *        STRS[DI] = 0 OR PS[DI] & PS[DJ] , ELIMINATE *XMT*               SQZ        1606     A
SQZ                                                                               SQZ        1607     A
SQZ        CTI3    SA2    "RN"+B6                                                  SQZ        1608     A
SQZ                SX7    B1                                                       SQZ        1609     A
SQZ                LX7    RN.NRP                                                   SQZ        1610     A
SQZ                BX2    X7*X2                                                    SQZ        1611     A
SQZ                BX6    X2+X1       RND(RI) = RND(RJ) ! NR[RI]                   SQZ        1612     A
SQZ                SA6    A2                                                       SQZ        1613     A
SQZ                MX7    0                                                        SQZ        1614     A
SQZ                SX4    B4                                                       SQZ        1615     A
SQZ                SA7    A4          [DI] = 0                                     SQZ        1616     A
SQZ                BX6    X3+X4                                                    SQZ        1617     A
SQZ                LX6    D.STRSP                                                  SQZ        1618     A
SQZ                SA6    A3          DJ = DJ ! STRS[DI]                           SQZ        1619     A
SQZ                EQ     MSL1                                                     SQZ        1620     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                                                                                  SQZ        1621    A
SQZ      *         PRS[DI] AND STRS[DJ] = 0                                          SQZ        1622    A
SQZ      *         KEEP XMT IF RJ IS A *DEF* OR *KS*, OR A *LD* AND THE *RS* IS      SQZ        1623    A
SQZ      *         NOT TO A LD REGISTER .                                            SQZ        1624    A
SQZ                                                                                  SQZ        1625    A
SQZ       CTI4     SA2    CTIB                                                       SQZ        1626    A
SQZ                LX7    X3                                                         SQZ        1627    A
SQZ                BX6    X2*X3                                                      SQZ        1628    A
SQZ                NZ     X6,CTI5     IF PI[DJ] ! KS[DJ]                             SQZ        1629    A
SQZ                LX7    D.STRSP+59-D.LDP                                           SQZ        1630    A
SQZ                MX6    -SO.RNL                                                    SQZ        1631    A
SQZ                PL     X7,CTI3     IF ^LD[DJ]                                     SQZ        1632    A
SQZ                                                                                  CCG0041      10    A
SQZ       #DAL     IFNE   .DAL,0                                                     SQZ        1633    A
SQZ                LX7    D.LDP-D.L2P                                                SQZ        1634          I
          -CCG0041
SQZ                LX7    D.LDP-D.XMTP                                               CCG0041      11    A
SQZ                MI     X7,CTI4A    IF XMT[DJ]    */ LEVEL 0 *LD*                  CCG0041      12    A
SQZ                LX7    D.XMTP-D.L2P                                               CCG0041      13    A
SQZ                MI     X7,CTI3     IF L2[DJ]                                      SQZ        1635    A
SQZ                                                                                  CCG0041      14    A
SQZ       CTI4A    BSS    0                                                          CCG0041      15    A
SQZ       #DAL     ENDIF                                                             SQZ        1636    A
SQZ                                                                                  CCG0041      16    A
SQZ                SA2    A4+B2       R1RS = DI + 2                                  SQZ        1637    A
SQZ                MX7    5                                                          SQZ        1638    A
SQZ                LX2    -R1.SOP                                                    SQZ        1639    A
SQZ                BX6    -X6*X2      REGN = SORN[R1RS]                              SQZ        1640    A
SQZ                LX7    -1                                                         SQZ        1641    A
SQZ                SB5    X6                                                         SQZ        1642    A
SQZ                LX7    B5,X7                                                      SQZ        1643    A
SQZ                MI     X7,CTI3     IF 1 @ REGN @ 5  */ A LD REG                   SQZ        1644    A
SQZ                                                                                  SQZ        1645    A
SQZ      *         KEEP *XMT*                                                        SQZ        1646    A
SQZ                                                                                  SQZ        1647    A
SQZ       CTI5     SA2    "RN"+B6                                                    SQZ        1648          I
          -FCC1825X
SQZ                                                                                  FCC1825X      5    A
SQZ       CTI5     SA2    CTIC                                                       FCC1825X      6    A
SQZ                LX3    D.STRSP                                                    FCC1825X      7    A
SQZ                BX6    -X2+X3                                                     FCC1825X      8    A
SQZ                ZR     X6,CTI8     IF ZP[DJ] & (SZ[DJ] = 1) & (TY[DJ] = II)       FCC1825X      9    A
SQZ                SA2    "RN"+B6                                                    FCC1825X     10    A
SQZ                SA4    A5+B2       DI = R1 + 2   */ RESTORE X4                    FCC0603      93    A
SQZ                SX3    B1                                                         SQZ        1649    A
SQZ                LX3    RN.XMTP                                                    SQZ        1650    A
SQZ                BX6    X3+X2       XMT[RI] = 1                                    SQZ        1651    A
SQZ                LX3    RN.CP-RN.XMTP                                              CCG17       220    A
SQZ                BX3    X3*X1                                                      CCG17       221    A
SQZ                BX6    X6+X3       C[RI] = C[RJ]                                  CCG17       222    A
SQZ                SA6    A2                                                         SQZ        1652    A
SQZ                EQ     CTI                                                        SQZ        1653    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                                                                                  SQZ        1654    A
SQZ      *         *XMT* PRECEDES A STORE. ELIMINATE IF PRED CAN PRECEDE A STORE.    SQZ        1655    A
SQZ                                                                                  SQZ        1656    A
SQZ       CTI6     SA2    CTIA                                                       SQZ        1657    A
SQZ                BX6    X2*X3                                                      SQZ        1658    A
SQZ       #DAL     IFEQ   .DAL,0                                                     SQZ        1659    A
SQZ                NZ     X6,CTI5     IF LD[DJ] ! KS[DJ]                             SQZ        1660    A
SQZ       #DAL     ELSE                                                              SQZ        1661    A
SQZ                ZR     X6,CTI7     IF ^( LD[DJ] ! KS[DJ] )                        SQZ        1662    A
SQZ                BX2    X3                                                         SQZ        1663    A
SQZ                LX2    D.STRSP-1-D.L2P                                            SQZ        1664    A
SQZ                PL     X2,CTI5     IF ^L2[DJ]                                     SQZ        1665    A
SQZ                LX2    D.L2P-D.XMTP                                     001350    CCG35        68    A
SQZ                MI     X2,CTI5     IF XMT[DI]    */ LEVEL 0 LD              001360    CCG35        69    A
SQZ                LX2    D.XMTP-D.FPP                                               CCG0154       6    A
SQZ                PL     X2,CTI7     IF NOT FP[DJ] */ ELIMINATE XMT                 CCG0154       7    A
SQZ                SA2    =XHO$OPT                                                   CCG0154       8    A
SQZ                MI     X2,CTI5     IF OPT .EQ. 2                                  CCG0154       9    A
SQZ                                                                                  CCG0154      10    A
SQZ      *         FP LOAD MAY CHANGE TO A LOAD OF A LC. , KEEP THE XMT              CCG0154      11    A
SQZ                                                                                  CCG0154      12    A
SQZ       CTI7     BSS    0                                                          SQZ        1666    A
SQZ       #DAL     ENDIF                                                             SQZ        1667    A
SQZ                BX2    X3                                                         SQZ        1668    A
SQZ                LX2    D.STRSP-1-D.PIP                                            SQZ        1669    A
SQZ                PL     X2,CTI3     IF ^PI[DJ]    */ RJ NOT A *DEF*                SQZ        1670    A
SQZ                                                                                  SQZ        1671    A
SQZ                SA2    A3-B2       R1J = DJ - 2                                   SQZ        1672    A
SQZ                MX6    -SO.RNL                                                    SQZ        1673    A
SQZ                LX2    -R1.SOP                                                    SQZ        1674    A
SQZ                BX7    -X6*X2                                                     SQZ        1675    A
SQZ                SX6    X7-6                                                       SQZ        1676    A
SQZ                MI     X6,CTI5     IF SORN[R1J] < 6  */ NOT A STORE REG           SQZ        1677    A
SQZ                SB4    B0          STRS[DI] = 0                                   SQZ        1678    A
SQZ                EQ     CTI3                                                       SQZ        1679    A
SQZ                                                                                  FCC1825X      11    A
SQZ      *         CLR OR FMA THAT PRS IN 2 REGS, DUPLICATE IT INSTEAD OF AN XMT     FCC1825X      12    A
SQZ                                                                                  FCC1825X      13    A
SQZ       CTI8     SA2    A3-B2                                                      FCC1825X      14    A
SQZ                SA4    A5+B2       DESCR OF XMT                                   FCC1825X      15    A
SQZ                BX6    X0*X2       OC AND IN OF R1J                               FCC1825X      16    A
SQZ                UX7    "OC",X2                                                    FCC1825X      17    A
SQZ                RJ     URI         CHANGE INSTRUCTION                             FCC1825X      18    A
SQZ                JP     SBB.JT+"OC"                                                FCC1825X      19    A
SQZ                                                                                  SQZ        1680    A
SQZ       CTIA     BFMW   D,(LD,KS),60-D.STRSP                                       SQZ        1681    A
SQZ       CTIB     BFMW   D,(KS,PI),60-D.STRSP                                       SQZ        1682    A
SQZ       CTIC     VFD    D.TYL/1     TYPE = II                                      FCC1825X      20    A
SQZ                POS    D.ZPP+1                                                    FCC1825X      21    A
SQZ                VFD    1/1                                                        FCC1825X      22    A
SQZ                POS    D.SZP+D.SZL                                                FCC1825X      23    A
SQZ                VFD    D.SZL/1,*P/0                                               FCC1825X      24    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ       STT      TITLE  MISCELLANEOUS INSTRUCTION PROCESSING                    SQZ        1683    A
SQZ      **        MISCELLANEOUS INSTRUCTION SQUEEZING                            SQZ        1684    A
SQZ      *         INSTRUCTIONS MAY BE REGISTER STORED TO X OR B REGISTERS.       SQZ        1685    A
SQZ      *         OR THEY ARE TYPE II ( NO PREDECESSORS ).                       SQZ        1686    A
SQZ                                                                               SQZ        1687    A
SQZ                PROCESS STT                                                    SQZ        1688    A
SQZ                SA2    A5+B1       R2 = R1 + 1                                 SQZ        1689    A
SQZ                LX2    -IH.RFP                                                SQZ        1690        I
          -FCC0792
SQZ                SA1    "RN"+X2     RFW = RND(RF[R2])                           SQZ        1691        I
          -FCC0792
SQZ                MX7    -RN.CIAL                                               SQZ        1692        I
          -FCC0792
SQZ                BX6    X0*X2       R2W = [R2]                                  SQZ        1693        I
          -FCC0792
SQZ                LX7    RN.CIAP                                                SQZ        1694        I
          -FCC0792
SQZ                BX3    -X7*X1                                                 SQZ        1695        I
          -FCC0792
SQZ                SX7    X1          RF[R2W] = RN[RFW]                           SQZ        1696        I
          -FCC0792
SQZ                ZR     X3,STT0     IF ^(C,SA,IAC)[RFW]                         SQZ        1697        I
          -FCC0792
SQZ                SA3    ARFA                                                   FCC0792       4    A
SQZ                LX2    -IH.RFP                                                FCC0792       5    A
SQZ                SA1    "RN"+X2     RFW = RND(RF[R2])                           FCC0792       6    A
SQZ                BX6    X0*X2       R2W = [R2]                                  FCC0792       7    A
SQZ                SX7    X1          RF[R2W] = RN[RFW]                           FCC0792       8    A
SQZ                BX3    X3*X1                                                  FCC0792       9    A
SQZ                ZR     X3,STT0     IF ^(C,SA,IAC,XMT)[RFW]                     FCC0792      10    A
SQZ                RJ     ARF         ADJUST RF,CA                                SQZ        1698    A
SQZ                                                                               SQZ        1699    A
SQZ       STT0     ZR     X6,STT5     IF CAIH[R2W] = 0  */ REDUCE TO SA           SQZ        1700    A
SQZ                BX6    X6+X7                                                  SQZ        1701    A
SQZ                SB4    X7          R = RF[R2W]                                 SQZ        1702    A
SQZ                LX6    IH.RFP                                                 SQZ        1703    A
SQZ                BX5    X6                                                     SQZ        1704    A
SQZ                SX7    X6                                                     SQZ        1705    A
SQZ                SA6    A5+B1       [R2] = R2W                                  SQZ        1706    A
SQZ                SB5    STT4                                                   SQZ        1707    A
SQZ                SA2    MIC                                                    SQZ        1708    A
SQZ                GT     B4,B1,STT2  IF R > 1                                   SQZ        1709    A
SQZ                NZ     B4,STT1     IF R " 0                                   SQZ        1710    A
SQZ                NZ     X7,STT1     IF IH[R2W] " 0                             SQZ        1711    A
SQZ                                                                               SQZ        1712    A
SQZ      *         RF & IH = 0 , CHANGE INST TO A *S*                             SQZ        1713    A
SQZ                                                                               SQZ        1714    A
SQZ                LX6    -IH.CAP                                                SQZ        1715    A
SQZ                SX7    X6          VAL = CA[R2]                                SQZ        1716    A
SQZ                MX6    0                                                      SQZ        1717    A
SQZ                SA6    A5+B1       [R2] = 0                                    SQZ        1718    A
SQZ                EQ     SA0                                                    SQZ        1719    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                                                                                  SQZ        1720    A
SQZ       STT1     SX3    B1                                                         SQZ        1721    A
SQZ                LX3    D.ZPP                                                      SQZ        1722    A
SQZ                BX7    X4+X3       ZP[DI] = 1                                     SQZ        1723    A
SQZ                SA7    A4                                                         SQZ        1724    A
SQZ                                                                                 SQZ        1725    A
SQZ       STT2     LX1    -RN.TPP                                                    SQZ        1726    A
SQZ                SB3    X1+3        DEFA = TP[RFW] + 3                             SQZ        1727    A
SQZ                SB4    X2          I = MIC                                        SQZ        1728    A
SQZ                SA2    X1+2        DF = TP[RFW] + 2                               SQZ        1729        I
          -FCC1928
SQZ                SX6    1                                                          SQZ        1730        I
          -FCC1928
SQZ                SX6    X5                                                         FCC1928       6    A
SQZ                ZR     X6,STT3     IF IH[R2] = 0 */ NO IH                         FCC1928       7    A
SQZ                                                                                 FCC1928       8    A
SQZ       .FTN     IFEQ   HC.ID,2                                          001050   CCG34        26    A
SQZ                SA2    =XS=TRACE                                       001060   CCG34        27    A
SQZ                BX7    X2-X6                                           001070   CCG34        28    A
SQZ                ZR     X7,STT3     IF IH[R2] = TRACE.  */ A-GOTO ERROR CHECK 001080 CCG34     29    A
SQZ       .FTN     ENDIF                                                   001090   CCG34        30    A
SQZ                                                                                 CCG0006       5    A
SQZ       .LCM     IFNE   .DAL,0                                                      CCG0006       6    A
SQZ                SA2    LCM#G                                                      CCG0006       7    A
SQZ                LX2    1+D.STP                                                    CCG0006       8    A
SQZ                BX7    X2*X4                                                      CCG0006       9    A
SQZ                NZ     X7,STT3     IF LCM = G AND L2[DI] /* 21 BIT ARITHMETIC    CCG0006      10    A
SQZ       .LCM     ENDIF                                                              CCG0006      11    A
SQZ                                                                                 CCG0006      12    A
SQZ                SA2    X1+B2       DF = TP[RFW] + 2                               FCC1928       9    A
SQZ                SX6    B1                                                         FCC1928      10    A
SQZ                LX6    D.RFP                                                      SQZ        1731    A
SQZ                BX7    X6+X2       RF[DF] = 1    */ SET SHORT USE FOR PRED        SQZ        1732    A
SQZ                SA7    A2+                                                        SQZ        1733        I
          -FCC1928
SQZ                SA7    A2                                                         FCC1928      11    A
SQZ                                                                                 SQZ        1734    A
SQZ       *        SEARCH *MIC* FOR PREVIOUS OCCURANCE OF THE *STT*                   SQZ        1735    A
SQZ                                                                                 SQZ        1736    A
SQZ       STT3     LE     B4,B3,SMI5  IF I @ DEFA                                    SQZ        1737    A
SQZ                SA1    B4-B2       R2W = [I-2]                                    SQZ        1738    A
SQZ                BX6    X5-X1                                                      SQZ        1739    A
SQZ                SA2    B4          I = [I]                                        SQZ        1740    A
SQZ                SB4    X2                                                         SQZ        1741    A
SQZ                NZ     X6,STT3     IF [R2W] " [R2]                                SQZ        1742    A
SQZ                                                                                 SQZ        1743    A
SQZ                SA2    A1-B1       R1W = R2W - 1                                  SQZ        1744    A
SQZ                SA3    A5          R1 = R1                                        SQZ        1745    A
SQZ                BX6    X2-X3                                                      SQZ        1746    A
SQZ                BX7    X0*X6                                                      SQZ        1747    A
SQZ                NZ     X7,STT3     IF (OC,H2)[R1] " (OC,H2)[R1W]                  SQZ        1748    A
SQZ                                                                                 SQZ        1749    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                SA3    A1+B1       DJ = R2W + 1                                SQZ        1750    A
SQZ                EQ     SMI2        GO TRY TO ELIMINATE IT                      SQZ        1751    A
SQZ                                                                               SQZ        1752    A
SQZ       STT4     LX5    R1.RJP-IH.RFP     INST ^ REDUNDANT                      SQZ        1753    A
SQZ                EQ     AUC.RJ                                                  SQZ        1754    A
SQZ                EQ     STT3        RETRY ADDRESS                               SQZ        1755    A
SQZ                                                                               SQZ        1756    A
SQZ      *         CAIH = 0 , REDUCE *STT* TO *SA* OR *S*                          SQZ        1757    A
SQZ                                                                               SQZ        1758    A
SQZ       STT5     MX6    0           [R2] = 0                                    SQZ        1759    A
SQZ                SA6    A5+B1                                                   SQZ        1760    A
SQZ                ZR     X7,SA0      IF RF[R2W] = 0                              SQZ        1761    A
SQZ                                                                               SQZ        1762    A
SQZ                S"OC"  OC.SA                                                   SQZ        1763    A
SQZ                BX6    X7                                                      SQZ        1764    A
SQZ                SA1    "RN"+X7                                                 SQZ        1765    A
SQZ                LX6    R1.RJP                                                  SQZ        1766    A
SQZ                SA2    "RN"                                                    SQZ        1767    A
SQZ                RJ     URI         URI( SA  RI, RF[R2W] )                      SQZ        1768    A
SQZ       SA       SPACE  3,14                                                    SQZ        1769    A
SQZ                PROCESS SA                                                     SQZ        1770    A
SQZ                BX6    X4                                                      CCG2         20    A
SQZ                SB5    SA.JT                                                   CCG2         21    A
SQZ                LX6    59-D.TYP                                                CCG2         22    A
SQZ                MI     X6,SMI      IF TYPE[DI] = II  */ SA RI,A0               CCG2         23    A
SQZ                LX2    59-RN.ZP                                                SQZ        1771    A
SQZ                PL     X2,SRCH     IF ^Z[RK]                                   SQZ        1772    A
SQZ                SB3    X1+                                                     CCG2         24    A
SQZ                EQ     B3,B1,SAA0  IF RJ[R1] = 1                               CCG2         25    A
SQZ                                                                               SQZ        1773    A
SQZ                BX6    X1                                                      SQZ        1774    A
SQZ                LX1    -RN.TPP                                                 SQZ        1775    A
SQZ                SA2    X1+B2       DJP = TP[RJ] + 2                            SQZ        1776    A
SQZ                SX3    B1                                                      SQZ        1777         I
          -FCC1928
SQZ                SB3    X1+3        DEFA = TP[RJ] + 3                           SQZ        1778    A
SQZ                SB5    SA.JT                                                   SQZ        1779         I
          -CCG2
SQZ                LX6    59-RN.CP                                                SQZ        1780    A
SQZ                LX3    D.RFP                                                   SQZ        1781         I
          -FCC1928
SQZ                BX7    X3+X2       RF[DJP] = 1                                 SQZ        1782         I
          -FCC1928
SQZ                SA7    A2                                                      SQZ        1783         I
          -FCC1928
SQZ                PL     X6,SA1      IF ^C[RJ]                                   SQZ        1784    A
SQZ                SA3    X1          R1J = TP[RJ]                                SQZ        1785    A
SQZ                RJ     DCV         VAL = DVC(R1J)                              SQZ        1786    A
SQZ                SX7    X7+                                                     SQZ        1787    A
SQZ                                                                               SQZ        1788    A
SQZ      *         SA0 - REDUCE SHORT CONSTANT TO A *S* INSTRUCTION                SQZ        1789    A
SQZ                                                                               SQZ        1790    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ       SA0     MX2     -18                                                    SQZ        1791    A
SQZ               BX1     X7                                                     SQZ        1792    A
SQZ               S"OC"   OC.S                                                   SQZ        1793    A
SQZ               BX6     -X2*X7                                                 SQZ        1794    A
SQZ               LX6     R1.INP                                                 SQZ        1795    A
SQZ               RJ      URI         URI(  S  RI,VAL )                          SQZ        1796    A
SQZ               NZ      X1,.S       IF VAL " 0                                 SQZ        1797    A
SQZ               SX3     B1                                                     SQZ        1798    A
SQZ               LX3     D.SZP                                                  SQZ        1799    A
SQZ               IX6     X4-X3       SZ[DI] = 1                                 SQZ        1800    A
SQZ               BX4     X6                                                     SQZ        1801    A
SQZ               SA6     A4                                                     SQZ        1802    A
SQZ               EQ      .CLR        GO SET Z BIT FOR RI                        SQZ        1803    A
SQZ                                                                              SQZ        1804    A
SQZ                                                                              SQZ        1805    A
SQZ       SA1     LX6     RN.CP-RN.IACP                                          SQZ        1806    A
SQZ               PL      X6,SMI      IF ^IAC[RJ]                                SQZ        1807    A
SQZ               SA3     ARFAIL                                      002611     CCG37         2    A
SQZ               NZ      X3,SMI      IF SECOND TRIP                  002612     CCG37         3    A
SQZ                                                                              SQZ        1808    A
SQZ               LX2     59-D.CMP                                               SQZ        1809    A
SQZ               MI      X2,SA2      IF CM[DJP]    */ *IA*                      SQZ        1810    A
SQZ               SA3     A2-B2       R1P = DJP - 2                              SQZ        1811    A
SQZ               LX3     -R1.RJP                                                SQZ        1812    A
SQZ               BX6     -X0*X3                                                 SQZ        1813    A
SQZ               SA2     "RN"+X6     RNJ = RND(RJ[R1P])                         SQZ        1814    A
SQZ               LX2     59-RN.CP                                               SQZ        1815    A
SQZ               MI      X2,SMI      IF C[RNJ]     */ CON - I CASE              SQZ        1816    A
SQZ                                                                              SQZ        1817    A
SQZ       *       CHANGE INSTRUCTION TO AN *STT*, LET *ARF* ADD THE CA.          SQZ        1818    A
SQZ                                                                              SQZ        1819    A
SQZ        SA2    LX5     -R1.RJP                                                SQZ        1820    A
SQZ               BX7     -X0*X5                                                 SQZ        1821    A
SQZ               LX7     IH.RFP                                                 SQZ        1822    A
SQZ               SA7     A5+B1       R2 = R1 + 1;  [R2] = IHW(0,RJ[R1],0,0)     SQZ        1823    A
SQZ               SX6     0                                                      SQZ        1824    A
SQZ               S"OC"   OC.STT                                                 SQZ        1825    A
SQZ               RJ      URI         URI(  STT  RI,RJ  )                        SQZ        1826    A
SQZ               EQ      .STT                                                   SQZ        1827    A
SQZ                                                                              SQZ        1828    A
SQZ       *       SA NOT REDUNDANT, SET *SA* BIT                                 SQZ        1829    A
SQZ                                                                              SQZ        1830    A
SQZ        SA5    SX2     B1                                                     SQZ        1831    A
SQZ               LX2     RN.SAP                                                 SQZ        1832    A
SQZ               BX6     X2+X3       SA[RND(RI)] = 1                            SQZ        1833    A
SQZ               SA6     A3                                                     SQZ        1834    A
SQZ               EQ      AUC.RJ                                                 SQZ        1835    A
SQZ                                                                              SQZ        1836    A
SQZ        SA.JT  SA3     "RN"+B6                                                SQZ        1837    A
SQZ               EQ      SA5                                                    SQZ        1838    A
SQZ                                                                              SQZ        1839    A
SQZ               EQ      SMI1        RETRY                                      SQZ        1840    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                                                                                    CCG2          26     A
SQZ      *         SA  RI,A0 - CHANGE TO A TYPE II INST SO *BDT* AND *MCG*             CCG2          27     A
SQZ      *         HANDLE IT PROPERLY.                                                 CCG2          28     A
SQZ                                                                                    CCG2          29     A
SQZ       SAA0     SX7    B1                                                           CCG2          30     A
SQZ                LX7    D.ZPP                                                        CCG2          31     A
SQZ                BX4    X7+X4       ZP[DI] = 1                                       CCG2          32     A
SQZ                LX7    D.TYP-D.ZPP                                                  CCG2          33     A
SQZ                BX4    X7+X4       TYPE[DI] = II                                    CCG2          34     A
SQZ                BX7    X4                                                           CCG2          35     A
SQZ                SA7    A4+                                                          CCG2          36     A
SQZ                EQ     SMI                                                          CCG2          37     A
SQZ       CLR      SPACE  3,14                                                         SQZ         1841     A
SQZ      *         ZERO - INSTRUCTION REDUCES TO A ZERO                                SQZ         1842     A
SQZ      *         CHANGE IT TO A *CLR* AND TRY TO ELIMINATE IT                        SQZ         1843     A
SQZ                                                                                    SQZ         1844     A
SQZ       ZERO     S"OC"  OC.CLR                                                       SQZ         1845     A
SQZ                MX6    0                                                            SQZ         1846     A
SQZ                SA6    A5+B1       [R2] = 0                                         SQZ         1847     A
SQZ                RJ     URI         URI( CLR  RI,0 )                                 SQZ         1848     A
SQZ       CLR      SPACE  3                                                            SQZ         1849     A
SQZ                PROCESS CLR                                                         SQZ         1850     A
SQZ                SA3    "RN"+B6                                                      SQZ         1851     A
SQZ                MX6    RN.CZL                                                       SQZ         1852     A
SQZ                LX6    RN.CZL+RN.CZP  C[RI] = Z[RI] = 1                             SQZ         1853     A
SQZ                BX7    X6+X3                                                        SQZ         1854     A
SQZ                SA7    A3                                                           SQZ         1855     A
SQZ                SB3    A0+3        DEFA = TXT + 3                                   FCC1085       24     A
SQZ                SB5    FMA1                                                         FCC1085       25     A
SQZ                EQ     SMI                                                          FCC1085       26     A
SQZ       FMA      SPACE  3                                                            SQZ         1856     A
SQZ       FMA      PROCESS (FMA,S)                                                     SQZ         1857     A
SQZ                SA3    "RN"+B6                                                      SQZ         1858          I
          -FCC1085
SQZ                UX6    X5                                                           FCC1085       27     A
SQZ                AX6    R1.INP                                                       FCC1085       28     A
SQZ                NZ     X6,FMA0     IF IN[R1] " 0 */ VAL(RI) NOT 0                   FCC1085       29     A
SQZ                BX6    X4                                                           FCC1085       30     A
SQZ                LX6    59-D.PRSP                                                    FCC1085       31     A
SQZ                PL     X6,ZERO     IF ^PRS[DI]                                      FCC1085       32     A
SQZ                EQ     .CLR                     */ GO SET *Z* BIT                   FCC1085       33     A
SQZ                                                                                    FCC1085       34     A
SQZ       FMA0     SA3    "RN"+B6                                                      FCC1085       35     A
SQZ                MX6    1                                                            SQZ         1859     A
SQZ                LX6    1+RN.CP                                                      SQZ         1860     A
SQZ                BX7    X6+X3       C[RI] = 1                                        SQZ         1861     A
SQZ                SA7    A3                                                           SQZ         1862     A
SQZ                SB3    A0+3        DEFA = TXT + 3                                   SQZ         1863     A
SQZ                SB5    FMA1                                                         SQZ         1864     A
SQZ                EQ     SMI                                                          SQZ         1865     A
SQZ                                                                                    SQZ         1866     A
SQZ       FMA1     EQ     MSL1        FAIL EXIT                                        SQZ         1867     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                EQ     SMI1        RETRY                                    SQZ        1868     A
SQZ       FMA60    SPACE  3                                                    SQZ        1869     A
SQZ      *         FMA60 - CHANGE INSTRUCTION TO  FMA  RI,60                   SQZ        1870     A
SQZ                                                                            SQZ        1871     A
SQZ       FMA60    S"OC"  OC.FMA                                               SQZ        1872     A
SQZ                SX6    60                                                   SQZ        1873     A
SQZ                LX6    R1.INP                                               SQZ        1874     A
SQZ                RJ     URI         URI( FMA  RI,60 )                        SQZ        1875     A
SQZ                EQ     FMA                                                  SQZ        1876     A
SQZ       LDC      SPACE  3,14                                                 CCG17       223     A
SQZ      **        LDC - LOAD CONSTANT ( LD  RI,0,CA,CON. )                    CCG17       224     A
SQZ                                                                            CCG17       225     A
SQZ                PROCESS LDC                                                 CCG17       226     A
SQZ                SA3    A5+B1       R2 = R1 + 1                              CCG17       227     A
SQZ                SA2    O.CVT                                                CCG17       228     A
SQZ                LX3    -IH.CAP                                              CCG17       229     A
SQZ                SB4    X3                                                   CCG17       230     A
SQZ                SA1    X2+B4                                                CCG17       231     A
SQZ                BX6    X1          VAL = CVT(CA[R2])                        CCG17       232     A
SQZ                SX7    X3                                                   CCG17       233     A
SQZ                LX7    I.CAP                                                CCG17       234     A
SQZ                SA7    A4+B1       CA[DI+1] = CA[R2]                        CCG17       235     A
SQZ                EQ     CCV         ATTEMPT TO REDUCE VAL TO A SHORT CONST.  CCG17       236     A
SQZ                                                                            CCG17       237     A
SQZ      *         CCV RETURNS TO LDC1 IF LDC INSTR. NOT REDUCED TO            CCG17       238     A
SQZ      *         *CLR*,*S* OR *FMA* INSTR.                                   CCG17       239     A
SQZ                                                                            CCG17       240     A
SQZ       LDC1     SA3    LDCC                                                 CCG17       241     A
SQZ                SA6    A5+B1       R2 = CONVAL   */ SAVE CONSTANT IN R2 WORD CCG17       242     A
SQZ                SB4    X3          I = LDCC                                 CCG17       243     A
SQZ                SA1    LDCN                                                 CCG17       244     A
SQZ                SX7    X1                                                   CCG17       245         I
          -CCG18
SQZ                SX7    X1          LDCN = SETX(LDCN)  */ CLEAR SIGN BIT     CCG18       160     A
SQZ                SA7    A1                                                   CCG17       246     A
SQZ                                                                            CCG17       247     A
SQZ       LDC2     SA2    B4          L = [I]                                  CCG17       248     A
SQZ                ZR     B4,LDC4     IF I = 0      */ END OF CHAIN            CCG17       249     A
SQZ                SA3    A2-B2       R2P = [L-2]                              CCG17       250     A
SQZ                BX7    X3-X6                                                CCG17       251     A
SQZ                SB4    X2          I = IC[L]                                CCG17       252     A
SQZ                NZ     X7,LDC2     IF NO MATCH                              CCG17       253     A
SQZ                MI     X7,LDC2     IF COMPLEMENTS                           CCG17       254     A
SQZ                                                                            CCG17       255     A
SQZ      *         MATCH FOUND, TRY TO ELIMINATE THE CURRENT LDC               CCG17       256     A
SQZ                                                                            CCG17       257     A
SQZ                SA3    A2-B1       DIM = LIP-1                              CCG17       258     A
SQZ                BX6    X3*X4                                                CCG17       259     A
SQZ                LX6    59-D.PRSP                                            CCG17       260     A
SQZ                BX7    X3+X4                                                CCG17       261     A
SQZ                MI     X6,LDC3     IF PRS[DIM] & PRS[DI]                    CCG17       262     A
SQZ                MX6    -D.STRSL                                             CCGA085       6     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ              LX7    -D.STRSP                                          CCGA085      7    A
SQZ              BX6    X6+X7                                             CCGA085      8    A
SQZ              LX7    D.STRSP                                           CCGA085      9    A
SQZ              ZR     X6,LDC3     IF PS AND PRS */ IXI CANT HANDLE      CCGA085     10    A
SQZ                                                                      CCG17      263    A
SQZ              SA7    A3          DIM = DIM ! DI                        CCG17      264    A
SQZ              MX6    0                                                 CCG17      265    A
SQZ              SA6    A4          DI = 0        */ KILL CURRENT LDC     CCG17      266    A
SQZ              SA2    A3-B2       R1M = DIM - 2                         CCG17      267    A
SQZ              SA1    "RN"+X2                                           CCG17      268    A
SQZ              BX7    X1                                                CCG17      269    A
SQZ              SA7    "RN"+B6     RND(RI[R1]) = RND(RI[R1M])            CCG17      270    A
SQZ              EQ     MSL1                                              CCG17      271    A
SQZ                                                                      CCG17      272    A
SQZ      *       BOTH PRS, GO CHANGE TO AN *XMT*                         CCG17      273    A
SQZ                                                                      CCG17      274    A
SQZ       LDC3   SA1    A3-B1                                             CCG17      275         I
         -CCG20
SQZ       LDC3   SA1    A3-B2                                             CCG20        1    A
SQZ              BX2    -X0*X1      RJ = RI[R1M]                          CCG17      276    A
SQZ              SA1    "RN"+X2     X1 = RND(RJ)  */FOR CTI               CCG17      277    A
SQZ              MX7    0                                                 CCG17      278    A
SQZ              SA7    A5+B1       [R2] = 0                              CCG17      279    A
SQZ              SA7    A4+B1       [DI+1] = 0                            CCG17      280    A
SQZ              SX2    X1          X2 = RN[RND(RJ)]                      CCG17      281    A
SQZ              EQ     SRCH6                                             CCG17      282    A
SQZ                                                                      CCG17      283    A
SQZ      *       LDC NOT REDUNDANT, ADD TO THE CHAIN                     CCG17      284    A
SQZ                                                                      CCG17      285    A
SQZ       LDC4   SA3    "RN"+B6                                           CCG17      286    A
SQZ              MX7    1                                                 CCG17      287    A
SQZ              LX7    1+RN.CP                                           CCG17      288    A
SQZ              BX7    X7+X3                                             CCG17      289    A
SQZ              SA7    A3          C[RND(RI)] = 1                        CCG17      290    A
SQZ              SA2    A4+B1                                             CCG17      291    A
SQZ              SA3    LDCC                                              CCG17      292    A
SQZ              BX6    X2+X3       IC[LI] = LDCC                         CCG17      293    A
SQZ              SA6    A4+B1                                             CCG17      294    A
SQZ              SX7    A4+B1       LDCC = LI     */ CHAIN POINTS TO LAST CCG17      295    A
SQZ              SA7    A3                                                CCG17      296    A
SQZ              NZ     X2,MSL1     IF CA[LI] " 0 */ NOT GENERATED BY SQZ CCG17      297    A
SQZ              SA3    LDCN                                              CCG17      298    A
SQZ              SX7    X3+B1                                             CCG17      299    A
SQZ              SA7    A3          LDCN = LDCN + 1                       CCG17      300    A
SQZ              EQ     MSL1                                              CCG17      301    A
SQZ       SMI    EJECT                                                    SQZ       1877    A
SQZ      **      SMI - SQUEEZE MISCELLANEOUS INSTRUCTIONS                 SQZ       1878    A
SQZ      *                                                                SQZ       1879    A
SQZ      *       ENTRY  (B3) = MIN SEARCH ADDRESS                         SQZ       1880    A
SQZ      *              (B5) = FAIL EXIT PROCESSOR ADDRESS                SQZ       1881    A
SQZ      *              X4,X5,B6 AS FROM MAIN LOOP                        SQZ       1882    A
SQZ      *       NOTE - IN CASE OF RETRY THIS SUBROUTINE MAY BE ENTERED   SQZ       1883    A

         0        1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ      *         AGAIN AT SMI1 / SMI2                                          SQZ        1884     A
SQZ                                                                              SQZ        1885     A
SQZ       SMI      SA2    MIC                                                   SQZ        1886     A
SQZ                SB4    X2          I = MIC                                   SQZ        1887     A
SQZ                                                                              SQZ        1888     A
SQZ      *         SEARCH *MIC* FOR A PREVIOUS OCCURANCE OF CURRENT INSTRUCTION. SQZ        1889     A
SQZ                                                                              SQZ        1890     A
SQZ       SMI1     LE     B4,B3,SMI5  IF I @ DEFA                               SQZ        1891     A
SQZ                SA1    B4-3                                                  SQZ        1892     A
SQZ                BX6    X5-X1                                                 SQZ        1893     A
SQZ                SA2    B4          I = [I]                                   SQZ        1894     A
SQZ                BX7    X0*X6                                                 SQZ        1895     A
SQZ                SB4    X2                                                    SQZ        1896     A
SQZ                NZ     X7,SMI1     IF MASK(-R1.RIL) & XOR(R1W,R1) " 0        SQZ        1897     A
SQZ                SA3    A1+B2       DJ = R1W + 2                              SQZ        1898     A
SQZ                                                                              SQZ        1899     A
SQZ      *         ELIMINATE REDUNDANT INSTRUCTION , (X3) = [DJ] + ABOVE REG ASSI SQZ        1900     A
SQZ                                                                              SQZ        1901     A
SQZ       SMI2     BX6    X3+X4                                                 SQZ        1902     A
SQZ                SX1    B4                                                    SQZ        1903     A
SQZ                MX7    -D.STRSL                                              SQZ        1904     A
SQZ                LX6    -D.STRSP                                              SQZ        1905     A
SQZ                BX6    -X7*X6                                                SQZ        1906     A
SQZ                SB4    X6                                                    SQZ        1907     A
SQZ                BX6    X3*X4                                                 SQZ        1908     A
SQZ                LX6    -D.STRSP                                              SQZ        1909     A
SQZ                BX7    -X7*X6                                                SQZ        1910     A
SQZ                JP     SMI3+B4     JUMP( STRS[DI] ! STRS[DJ] )               SQZ        1911     A
SQZ                                                                              SQZ        1912     A
SQZ       SMI3     EQ     SMI6        STRS[DJ] & STRS[DI] = 0                   SQZ        1913     A
SQZ                                                                              SQZ        1914     A
SQZ                NZ     X7,SMI7     IF PRS[DJ] & PRS[[DI]                     SQZ        1915     A
SQZ                EQ     SMI8                                                  SQZ        1916     A
SQZ                                                                              SQZ        1917     A
SQZ                EQ     SMI6        PS[DI] ! PS[DJ]                           SQZ        1918     A
SQZ                                                                              SQZ        1919     A
SQZ       SMI3A    SB4    X1+                                                   SQZ        1920     A
SQZ      *         EQ     SMI4        PS & PRS                                  SQZ        1921     A
SQZ                                                                              SQZ        1922     A
SQZ       SMI4     ZR     B4,SMI5     IF I = 0                                  SQZ        1923     A
SQZ                JP     B5+1        */ EXIT FOR A RETRY                       SQZ        1924     A
SQZ                                                                              SQZ        1925     A
SQZ      *         INSTRUCTION NOT REDUNDANT, RETURN TO FAIL EXIT                SQZ        1926     A
SQZ                                                                              SQZ        1927     A
SQZ       SMI5     SA2    MIC                                                   SQZ        1928     A
SQZ                BX6    X2                                                    SQZ        1929     A
SQZ                SA6    A4+B1       [DI+1] = MIC                              SQZ        1930     A
SQZ                SX7    A4+B1                                                 SQZ        1931     A
SQZ                SA7    A2          MIC = DI + 1                              SQZ        1932     A
SQZ                JP     B5          */ RETURN TO FAIL EXIT                    SQZ        1933     A
SQZ                                                                              SQZ        1934     A
SQZ      *         INSTRUCTION REDUNDANT, ELIMINATE SECOND OCCURANCE             SQZ        1935     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                                                                                  SQZ        1936     A
SQZ       SMI6     SA2    A3-B2       R1J = DJ - 2                                   SQZ        1937     A
SQZ                BX6    X3+X4                                                      SQZ        1938     A
SQZ                SA6    A3          DJ = DJ ! DI                                   SQZ        1939     A
SQZ                BX1    -X0*X2                                                     SQZ        1940     A
SQZ                MX7    0                                                          SQZ        1941     A
SQZ                SA2    "RN"+X1                                                    SQZ        1942     A
SQZ                BX6    X2                                                         SQZ        1943     A
SQZ                SA7    A4          [DI] = 0      */ MARK INST AS DEAD             SQZ        1944     A
SQZ                SA6    "RN"+B6     RND(RI) = RND(RI[R1J])                         SQZ        1945     A
SQZ                EQ     MSL1                                                       SQZ        1946     A
SQZ                                                                                  SQZ        1947     A
SQZ      *         BOTH INSTRUCTIONS PRECEDE A REGISTER STORE.                       SQZ        1948     A
SQZ                                                                                  SQZ        1949     A
SQZ       SMI7     SB4    X1                                                         SQZ        1950     A
SQZ                SA1    A3+B1                                                      SQZ        1951     A
SQZ                SA2    A4+B2       RSI = R1 + 4                                   SQZ        1952     A
SQZ                LX1    -36                                                        SQZ        1953          I
          -CCG17
SQZ                LX1    -I.RSP                                                     CCG17       302     A
SQZ                SA1    X1          RSJ = [RSL[DJ+1]]                              SQZ        1954     A
SQZ                BX6    X1-X2                                                      SQZ        1955     A
SQZ                BX6    X0*X6                                                      SQZ        1956     A
SQZ                NZ     X6,SMI4     IF REG[RSJ] " REG[RSI]                         SQZ        1957     A
SQZ                                                                                  SQZ        1958     A
SQZ                MX7    60                                                         SQZ        1959     A
SQZ                SA7    A4+4        [DI+4] = -0   */ MARK FOLLOWING *RS* AS DEA    SQZ        1960     A
SQZ                EQ     SMI6                                                       SQZ        1961     A
SQZ                                                                                  SQZ        1962     A
SQZ      *         ONE OF THE INSTRUCTIONS PRECEDES AN RS.  IF IT IS TO A B REG,     SQZ        1963     A
SQZ      *         WE CANNOT REDUCE IT.                                              SQZ        1964     A
SQZ                                                                                  SQZ        1965     A
SQZ       SMI8     LX4    59-D.PRSP                                                  SQZ        1966     A
SQZ                SA2    A4+B1       R1RS = DI + 1                                  SQZ        1967     A
SQZ                MI     X4,SMI9     IF PRS[DI]                                     SQZ        1968     A
SQZ                SA2    A3+B1                                                      SQZ        1969     A
SQZ                LX2    -36                                                        SQZ        1970     A
SQZ                SA2    X2          R1RS = [RSL[DJ+1]]                             SQZ        1971     A
SQZ                                                                                  SQZ        1972     A
SQZ       SMI9     LX4    1+D.PRSP                                                   SQZ        1973     A
SQZ                MX6    -SO.RTL                                                    SQZ        1974     A
SQZ                LX2    -R1.SOP-SO.RTP                                             SQZ        1975     A
SQZ                BX7    -X6*X2                                                     SQZ        1976     A
SQZ                NZ     X7,SMI6     IF SORT[R1RS] " 0  */ ^ A B-REG                SQZ        1977     A
SQZ                EQ     SMI3A                                                      SQZ        1978     A
SQZ       URI      EJECT                                                             SQZ        1979          I
          -CCG17
SQZ       URI      TITLE  COMPILE TIME INSTRUCTION EVALUATION                        CCG17       303     A
SQZ      **        URI - UPDATE REDUCED INSTRUCTION                                  SQZ        1980     A
SQZ      *         UPDATES CORE IMAGE AND REGISTERS OF A CHANGED TYPE I/II INST      SQZ        1981     A
SQZ      *                                                                           SQZ        1982     A
SQZ      *         ENTRY  (B3) = OPCODE                                              SQZ        1983     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ      *                (X4) = OLD DESCRIPTOR                                    SQZ        1984     A
SQZ      *                (X6) = RJ/RK OR IN/SO FIELDS OF NEW WORD 1               SQZ        1985     A
SQZ      *                                                                         SQZ        1986     A
SQZ      *         EXIT   (X4) = [DI] = ( NEW )                                     SQZ        1987     A
SQZ      *                (X5) = [R1] ( NEW )                                      SQZ        1988     A
SQZ      *                                                                         SQZ        1989     A
SQZ      *         USES   X - 3, 6, 7                                               SQZ        1990     A
SQZ      *                A - 3, 6, 7                                              SQZ        1991     A
SQZ                                                                                SQZ        1992     A
SQZ       URI      ROUTINE                                                          SQZ        1993     A
SQZ                SA3    F.RDT+"OC"                                                SQZ        1994     A
SQZ                MX7    -D.STRSL                                                  SQZ        1995     A
SQZ                SX5    B6                                                        SQZ        1996     A
SQZ                LX7    D.STRSP                                                   SQZ        1997     A
SQZ                IX6    X6+X5                                                     SQZ        1998     A
SQZ                BX4    -X7*X4                                                    SQZ        1999     A
SQZ                PX6    "OC",X6     [R1] = NEW WORD 1                             SQZ        2000     A
SQZ                SA6    A5                                                        SQZ        2001     A
SQZ                BX7    X3+X4       [DI] = RDT(OC) ! STRS[DI]                     SQZ        2002     A
SQZ                LX5    X6                                                        SQZ        2003     A
SQZ                SA7    A4                                                        SQZ        2004     A
SQZ                LX4    X7                                                        SQZ        2005     A
SQZ                EQ     URI                                                       SQZ        2006     A
SQZ       ECO      SPACE  2,14                                                      SQZ        2007     A
SQZ      **        ECO - EVALUATE INSTRUCTION WITH CONSTANT OPERANDS                SQZ        2008          I
         -CCG17
SQZ      *                                                                         SQZ        2009          I
         -CCG17
SQZ      *         ENTRY  (B3) = INSTRUCTION OPCODE ( = MACHINE OPCODE )            SQZ        2010          I
         -CCG17
SQZ                                                                                SQZ        2011          I
         -CCG17
SQZ       ECO      SA3    ECOA                                                      SQZ        2012          I
         -CCG17
SQZ                SX6    "OC"                                                      SQZ        2013          I
         -CCG17
SQZ                LX6    54                                                        SQZ        2014          I
         -CCG17
SQZ                BX7    X6+X3                                                     SQZ        2015          I
         -CCG17
SQZ                SA7    ECO1                                                      SQZ        2016          I
         -CCG17
SQZ                RJ     GCV         GET CONSTANT VALUES ( ALSO FLUSH STACK )      SQZ        2017          I
         -CCG17
SQZ                SB4    20B                                                       SQZ        2018          I
         -CCG17
SQZ                LT     B3,B4,ECO1  IF OC[R1] < 20B   */ BOOLEAN INSTRUCTION     SQZ        2019          I
         -CCG17
SQZ                OR     X6,SRCH     IF ^RANGE(VAL[RJ])                            SQZ        2020          I
         -CCG17
SQZ                ID     X6,SRCH     IF INDEF(VAL[RJ])                             SQZ        2021          I
         -CCG17

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                OR     X7,SRCH     IF ^RANGE(VAL[RK])                          SQZ        2022        I
           -CCG17
SQZ                ID     X7,SRCH     IF INDEF(VAL[RK])                           SQZ        2023        I
           -CCG17
SQZ        ECO1    FX6    X6+X7       VAL = VAL[RJ] .OP. VAL[RK]                  SQZ        2024        I
           -CCG17
SQZ                LT     B3,B4,CCV   IF OC[R1] < 20B                             SQZ        2025        I
           -CCG17
SQZ                                                                               SQZ        2026        I
           -CCG17
SQZ        +       OR     X6,SRCH     IF ^RANGE(VAL)                              SQZ        2027        I
           -CCG17
SQZ                ID     X6,SRCH     IF INDEF(VAL)                               SQZ        2028        I
           -CCG17
SQZ                EQ     CCV                                                     SQZ        2029        I
           -CCG17
SQZ                                                                               SQZ        2030        I
           -CCG17
SQZ        ECOA    VFD    6/00B,9/667B                                            SQZ        2031        I
           -CCG17
SQZ                LT     B3,B4,CCV                                               SQZ        2032        I
           -CCG17
SQZ                NO                                                             SQZ        2033        I
           -CCG17
SQZ        **      ECO PROCESSORS - EVALUATE INSTRUCTION WITH CONSTANT OPERANDS   CCG17       304     A
SQZ        *                                                                      CCG17       305     A
SQZ        *       ENTRY  CURRENT INSTRUCTIONS IS A TYPE I INSTR AND              CCG17       306     A
SQZ        *       X1,X2,X4,X5,B3,B6 ARE AS SET IN THE MAIN LOOP                  CCG17       307     A
SQZ        *                                                                      CCG17       308     A
SQZ                                                                               CCG17       309     A
SQZ                EPROC  IM                                                      CCG17       310     A
SQZ                S"OC"  OC.DFM                                                  CCG17       311     A
SQZ                                                                               CCG17       312     A
SQZ                EPROC  (FA,FS,DFA,DFS,RFA,RFS,FM,RFM,DFM,FD,RFD)               CCG17       313     A
SQZ         ECO1   SA3    ECOA                                                   CCG17       314     A
SQZ                SX6    "OC"                                                   CCG17       315     A
SQZ                LX6    54          */ SET OP                                  CCG17       316     A
SQZ                BX7    X6+X3                                                  CCG17       317     A
SQZ                SA7    ECO2        */ PLUG CODE FOR CURR."OC" AT ECO2         CCG17       318     A
SQZ                RJ     GCV         X6 = JVAL                                  CCG17       319     A
SQZ        *                          X7 = KVAL                                  CCG17       320     A
SQZ                OR     X6,SRCH     IF ^RANGE(JVAL)                            CCG17       321     A
SQZ                ID     X6,SRCH     IF  INDEF(JVAL)                            CCG17       322     A
SQZ                OR     X7,SRCH     IF ^RANGE(KVAL)                            CCG17       323     A
SQZ                ID     X7,SRCH     IF  INDEF(KVAL)                            CCG17       324     A
SQZ                SX1    "OC"-44B    X1 = PL IF DIVIDE INSTRUCTION              CCGA024       5     A
SQZ                UX3    X6,B4       B4 = EXPONENT OF 1OP                       CCGA024       6     A
SQZ                SB4    B4+60B-1771B-60B                                       CCGA024       7     A
SQZ                UX3    X7,B5       B5 = EXPONENT OF 2OP                       CCGA024       8     A
SQZ                SB5    B5+60B                                                 CCGA024       9     A
SQZ                MI     X1,ECO1A    IF NOT DIVIDE INSTRUCTION                  CCGA024      10     A
SQZ                NX3    X7                                                     CCGA024      11     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                ZR     X3,SRCH     IF DIVIDE BY ZERO                                CCGA024      12     A
SQZ                SB4    B4-B5                                                        CCGA024      13     A
SQZ                PL     B4,SRCH     IF RESULT OF DIVIDE WOULD BE BAD                 CCGA024      14     A
SQZ                EQ     ECO2                                                         CCGA024      15     A
SQZ                                                                                    CCGA024      16     A
SQZ       ECO1A    SX3    "OC"-40B    ADD AND SUBTRACT OP CODES ARE .LT. 40B           CCGA024      17     A
SQZ                MI     X3,ECO2     IF ADD OR SUBTRACT                               CCGA024      18     A
SQZ                SB4    B4+B5                                                        CCGA024      19     A
SQZ                PL     B4,SRCH     IF RESULT OF MULTIPLY WOULD BE BAD               CCGA024      20     A
SQZ                                                                                    CCG17       325     A
SQZ       ECO2     BX6    X6+X7       VAL = JVAL .OP. KVAL  */ EVALUATE INSTRUCTION    CCG17       326     A
SQZ                                                                                    CCG17       327     A
SQZ      +         ID     X6,SRCH     IF INDEF(VAL)                                    CCG17       328     A
SQZ                OR     X6,SRCH     IF ^RANGE(VAL)                                   CCGA024      21     A
SQZ                EQ     CCV         GO CHANGE A NEW CONSTANT                         CCG17       329     A
SQZ                                                                                    CCG17       330     A
SQZ       ECOA     VFD    6/0B,9/667B                                                  CCG17       331     A
SQZ                OR     X6,SRCH     IF ^RANGE(VAL)                                   CCG17       332           I
          -CCGA024
SQZ                                                                                    CCG17       333     A
SQZ                EPROC  CX                                                           CCG17       334     A
SQZ                BX2    X1                                                           CCG17       335     A
SQZ                EQ     ECO.AND                                                      CCG17       336     A
SQZ                                                                                    CCG0002       5     A
SQZ                EPROC  SS                                                           CCG0002       6     A
SQZ                RJ     GCV                                                          CCG0002       7     A
SQZ                SB4    X6                                                           CCG0002       8     A
SQZ                SB5    X7                                                           CCG0002       9     A
SQZ                SX6    B4-B5                                                        CCG0002      10     A
SQZ                EQ     CCV                                                          CCG0002      11     A
SQZ                                                                                    CCG17       337     A
SQZ                EPROC  (IAZ,ISZ)                                                    CCG17       338     A
SQZ                S"OC"  "OC"-OC.IAZ+OC.IA                                            CCG17       339     A
SQZ                                                                                    CCG17       340     A
SQZ                EPROC  (AND,OR,XOR,IA,IS)                                           CCG17       341     A
SQZ       ECO.AND  SX6    667B                                                         CCG17       342     A
SQZ                EQ     ECO3                                                         CCG17       343     A
SQZ                                                                                    CCG17       344     A
SQZ                EPROC  (STR,IMP,EQV)                                                CCG17       345     A
SQZ                SX6    676B                                                         CCG17       346     A
SQZ                EQ     ECO3                                                         CCG17       347     A
SQZ                                                                                    CCG17       348     A
SQZ                EPROC  (ILS,IRS,PK)                                                 CCG17       349     A
SQZ                SX6    657B                                                         CCG17       350     A
SQZ                EQ     ECO3                                                         CCG17       351     A
SQZ                                                                                    CCG17       352     A
SQZ                EPROC  (NR,RNZ,UP)                                                  CCG17       353           I
          -CCG18
SQZ       ECO.NR   EPROC  (NR,RNZ,UP)                                                  CCG18       161     A
SQZ                SX6    607B                                                         CCG18       162     A
SQZ       ECO3     SA3    ECOB                                                         CCG17       354     A
SQZ                SX7    "OC"                                                         CCG17       355     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                LX6    30                                          CCG17    356    A
SQZ                LX7    39                                          CCG17    357    A
SQZ                BX7    X6+X7                                       CCG17    358    A
SQZ                BX7    X3+X7                                       CCG17    359    A
SQZ                SA7    ECO3A                                       CCG17    360    A
SQZ                RJ     GCV         X6 = JVAL                       CCG17    361    A
SQZ      *                            X7 = KVAL                       CCG17    362    A
SQZ       ECO3A    SB5    X6          B5 = LOW18[JVAL]                CCG17    363    A
SQZ                NO                                                CCG17    364    A
SQZ                EQ     CCV                                         CCG17    365    A
SQZ                                                                  CCG17    366    A
SQZ       ECOB     SB5    X6                                          CCG17    367    A
SQZ                VFD    15/0                                        CCG17    368    A
SQZ                EQ     CCV                                         CCG17    369    A
SQZ                                                                  CCG17    370    A
SQZ      *         PROCESS NR,RNZ,UP WITH TWO RESULTS                 CCG17    371    A
SQZ                                                                  CCG17    372    A
SQZ       ECO4     SA3    ECOC                                        CCG17    373    A
SQZ                SX6    "OC"                                        CCG17    374    A
SQZ                LX6    54                                          CCG17    375    A
SQZ                BX7    X6+X3                                       CCG17    376    A
SQZ                SA7    ECO4A                                       CCG17    377    A
SQZ                LX2    -RN.TPP                                     CCG17    378    A
SQZ                SA3    X2                                          CCG17    379    A
SQZ                RJ     DCV         X7 = KVAL                       CCG17    380    A
SQZ                                                                  CCG17    381    A
SQZ       ECO4A    UX1    B5,X7       EVALUATE INSTRUCTION            CCG17    382    A
SQZ                SX6    B5                                          CCG17    383    A
SQZ                S"OC"  OC.S        OC = OC.S                       CCG17    384    A
SQZ                MX7    -18                                         CCG17    385    A
SQZ                BX6    -X7*X6                                      CCG17    386    A
SQZ                LX6    R1.INP      IN = VAL1                       CCG17    387    A
SQZ                SA5    A4+B2       R1RS = DI + 2                   CCG17    388    A
SQZ                MX7    -R1.SOL                                     CCG17    389    A
SQZ                LX7    R1.SOP                                      CCG17    390    A
SQZ                SX3    SO.TLOCK                                    CCG17    391    A
SQZ                LX3    R1.SOP                                      CCG17    392    A
SQZ                BX7    -X7*X5                                      CCG17    393    A
SQZ                BX7    X7+X3       SO = SO[R1RS] + SO.TLOCK        CCG17    394    A
SQZ                BX6    X6+X7                                       CCG17    395    A
SQZ                SA4    A5+B2       DIRS = R1RS + 2                 CCG17    396    A
SQZ                BX3    -X0*X5                                      CCG17    397    A
SQZ                SB6    X3          RIRS = RI[R1RS]                 CCG17    398    A
SQZ                MX7    0                                           CCG17    399    A
SQZ                SA7    A5+B1       [R2RS] = 0                      CCG17    400    A
SQZ                RJ     URI         URI( S  R1RS,VAL1 )             CCG17    401    A
SQZ                SA4    A5-B2       DI = R1RS - 2                   CCG17    402    A
SQZ                SA5    A4-B2       R1 = DI - 2                     CCG17    403    A
SQZ                BX3    -X0*X5                                      CCG17    404    A
SQZ                SB6    X3          RI = RI[R1]                     CCG17    405    A
SQZ                BX6    X1                                          CCG17    406    A
SQZ                SX7    B1                                          CCG17    407    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                SA7    IXF                                                  CCG17      408    A
SQZ                EQ     CCV         CCV( LDC RI,VAL2 )                       CCG17      409    A
SQZ                                                                            CCG17      410    A
SQZ       ECOC     VFD    6/0,9/157B                                           CCG17      411    A
SQZ                SX6    B5                                                   CCG17      412    A
SQZ                S"OC"  OC.S                                                 CCG17      413    A
SQZ       GCV      SPACE  2,14                                                 SQZ       2034    A
SQZ      **        GCV - GET CONSTANT VALUES ( CONSTANT FOLDING )              SQZ       2035    A
SQZ      *                                                                     SQZ       2036    A
SQZ      *         ENTRY  (X1/X2) = RND(RJ/RK)                                 SQZ       2037    A
SQZ      *                                                                     SQZ       2038    A
SQZ      *         EXIT   (X6,X7) = VAL[RJ/RK]                                 SQZ       2039    A
SQZ                                                                            SQZ       2040    A
SQZ       GCV      ROUTINE                                                     SQZ       2041    A
SQZ                LX1    -RN.TPP                                              SQZ       2042    A
SQZ                SA3    X1          R1P = TP[RJ]                             SQZ       2043    A
SQZ                RJ     DCV                                                  SQZ       2044    A
SQZ                LX2    -RN.TPP                                              SQZ       2045    A
SQZ                BX6    X7          (X6) = VAL[RJ]                           SQZ       2046    A
SQZ                SA3    X2          R1P = TP[RK]                             SQZ       2047    A
SQZ                RJ     DCV         (X7) = VAL[RK]                           SQZ       2048    A
SQZ                EQ     GCV                                                  SQZ       2049    A
SQZ       DCV      SPACE  3,14                                                 SQZ       2050    A
SQZ      **        DCV - DETERMINE CONSTANT VALUE                              SQZ       2051    A
SQZ      *                                                                     SQZ       2052    A
SQZ      *         ENTRY  (X3) = R1 WORD OF *S*,*FMA* INSTRUCTION              SQZ       2053          I
         -CCG17
SQZ      *         ENTRY  (X3) = R1 WORD OF *S*,*FMA* OR *LDC* INSTRUCTION     CCG17      414    A
SQZ      *                (A3) = ADDRESS OF R1 WORD                            CCG17      415    A
SQZ      *                                                                     SQZ       2054    A
SQZ      *         EXIT   (X7) = VALUE OF CONSTANT                             SQZ       2055    A
SQZ      *                (B5) = 0 IF OC = OC.S                                SQZ       2056          I
         -CCG17
SQZ                                                                            SQZ       2057          I
         -CCG17
SQZ       DCV      ROUTINE                                                     SQZ       2058          I
         -CCG17
SQZ                UX3    B4,X3                                                SQZ       2059          I
         -CCG17
SQZ                SB5    B4-OC.S                                              SQZ       2060          I
         -CCG17
SQZ                LX3    -R1.INP                                              SQZ       2061          I
         -CCG17
SQZ                SX7    X3          VAL = IN[R1]                             SQZ       2062          I
         -CCG17
SQZ                ZR     B5,DCV      IF OC[R1] = OC.S                         SQZ       2063          I
         -CCG17
SQZ                SB4    X3-1                    */ ASSUME A *FMA*            SQZ       2064          I
         -CCG17
SQZ                MX3    1                                                    SQZ       2065          I
         -CCG17
SQZ                AX7    B4,X3       VAL = MASK(IN[R1])                       SQZ       2066          I

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

          -CCG17
SQZ                 PL     B4,DCV      IF IN[R1] \ 1                          SQZ        2067        I
          -CCG17
SQZ                 MX7    0           VAL = 0                                SQZ        2068        I
          -CCG17
SQZ                 EQ     DCV                                                SQZ        2069        I
          -CCG17
SQZ       *                (B5) = OPCODE  ( USED BY *XMTC* PROCESSOR )        CCG17       416     A
SQZ       *                                                                   CCG17       417     A
SQZ       *         USES   X3, X7, B4, B5                                     CCG17       418     A
SQZ                                                                           CCG17       419     A
SQZ        DCV0     RJ     GIV                                                CCG17       420     A
SQZ                                                                           CCG17       421     A
SQZ        DCV      ROUTINE                                                   CCG17       422     A
SQZ                                                                           CCG17       423     A
SQZ        DCV1     UX7    B4,X3                                              CCG17       424     A
SQZ                 SB5    OC.XMTC                                            CCG17       425     A
SQZ                 LX7    -R1.RJP                                            CCG17       426     A
SQZ                 EQ     B4,B5,DCV2  IF OC = OC.XMTC                        CCG17       427     A
SQZ                 SB5    OC.XMT                                             CCG17       428     A
SQZ                 NE     B4,B5,DCV0  IF OC " OC.XMT                         CCG17       429     A
SQZ                 BX7    -X0*X7                                             CCG17       430     A
SQZ                 SA3    "RN"+X7                                            CCG17       431     A
SQZ                 LX3    -RN.TPP                                            CCG17       432     A
SQZ                 SA3    X3                                                 CCG17       433     A
SQZ                 EQ     DCV1                                               CCG17       434     A
SQZ                                                                           CCG17       435     A
SQZ        DCV2     BX7    -X0*X7                                             CCG17       436     A
SQZ                 SA3    "RN"+X7                                            CCG17       437     A
SQZ                 LX3    -RN.TPP                                            CCG17       438     A
SQZ                 SA3    X3                                                 CCG17       439     A
SQZ                 RJ     GIV         GET VALUE                              CCG17       440     A
SQZ                 BX7    -X7         VAL = -VAL                             CCG17       441     A
SQZ                 EQ     DCV                                                CCG17       442     A
SQZ        GIV      SPACE  2,14                                               CCG17       443     A
SQZ        GIV0     SB4    X7-1                                               CCG17       444     A
SQZ                 ZR     X7,GIV      IF IN[R1] = 0                          CCG17       445     A
SQZ                 MX7    1                                                  CCG17       446     A
SQZ                 AX7    B4          VAL = MASK(IN[R1])                     CCG17       447     A
SQZ        GIV      ROUTINE                                                   CCG17       448     A
SQZ                                                                           CCG17       449     A
SQZ        GIV1     UX3    B4,X3                                              CCG17       450     A
SQZ                 SB5    OC.CLR                                             CCG17       451     A
SQZ                 MX7    0           VAL = 0                                CCG17       452     A
SQZ                 EQ     B4,B1,GIV   IF OC.BOS                              CCG17       453     A
SQZ                 EQ     B4,B5,GIV   IF OC = OC.CLR                         CCG17       454     A
SQZ                 LX3    -R1.INP                                            CCG17       455     A
SQZ                 SX7    X3          VAL = IN[R1]                           CCG17       456     A
SQZ                 SB5    OC.S                                               CCG17       457     A
SQZ                 EQ     B4,B5,GIV   IF OC = OC.S                           CCG17       458     A
SQZ                 SB5    OC.FMA                                             CCG17       459     A
SQZ                 EQ     B4,B5,GIV0  IF OC = OC.FMA                         CCG17       460     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                SB5    OC.LDC                                          CCG17        461     A
SQZ                NE     B4,B5,GIV2  IF OC " OC.LDC                      CCG17        462     A
SQZ                SA3    A3+B1                                           CCG17        463     A
SQZ                BX7    X3          VAL = [R2]                          CCG17        464     A
SQZ                EQ     GIV                                             CCG17        465     A
SQZ                                                                       CCG17        466     A
SQZ        GIV2    SB5    OC.XMT                                          CCG17        467     A
SQZ                NE     B4,B5,*+1S17                                    CCG17        468     A
SQZ                LX3    R1.INP-R1.RJP                                   CCG17        469     A
SQZ                BX7    -X0*X3                                          CCG17        470     A
SQZ                SA3    "RN"+X7                                         CCG17        471     A
SQZ                LX3    -RN.TPP                                         CCG17        472     A
SQZ                SA3    X3                                              CCG17        473     A
SQZ                EQ     GIV1                                            CCG17        474     A
SQZ        CCV     SPACE  3                                               SQZ         2070     A
SQZ       **       CCV - CONVERT CONSTANT VALUE                           SQZ         2071     A
SQZ       *                                                               SQZ         2072     A
SQZ       *        ENTRY  (X6) = VAL , TO BE REFORMED AS *S*/*FMA* INSTRUCTION  SQZ  2073          I
          -CCG17
SQZ       *        ENTRY  (X6) = VAL                                      CCG17        475     A
SQZ       *               (B6) = RI                                       SQZ         2074     A
SQZ                                                                       SQZ         2075     A
SQZ       *        TRY TO REDUCE CONSTANT TO A *FMA*                      SQZ         2076     A
SQZ                                                                       SQZ         2077     A
SQZ        CCV     PL     X6,CCV1     IF VAL > 0                          SQZ         2078     A
SQZ                CX7    X6                                              SQZ         2079     A
SQZ                MX3    1                                               SQZ         2080     A
SQZ                SB4    X7-1                                            SQZ         2081     A
SQZ                AX7    B4,X3                                           SQZ         2082     A
SQZ                BX3    X7-X6                                           SQZ         2083     A
SQZ                NZ     X3,CCV1     IF VAL " MASK( COUNT(VAL) )         SQZ         2084     A
SQZ                S"OC"  OC.FMA      OC = OC.FMA                         SQZ         2085     A
SQZ                SX6    B4+B1       VAL = COUNT(VAL)                    SQZ         2086     A
SQZ                EQ     CCV2                                            SQZ         2087     A
SQZ                                                                       SQZ         2088     A
SQZ       *        TRY TO REDUCE CONSTANT TO A *S*                        SQZ         2089     A
SQZ                                                                       SQZ         2090     A
SQZ        CCV1    ZR     X6,ZERO     IF VAL = 0                          SQZ         2091     A
SQZ                MX3    42                                              SQZ         2092          I
          -CCG17
SQZ                LX7    X6                                              SQZ         2093     A
SQZ                BX6    -X3*X6                                          SQZ         2094          I
          -CCG17
SQZ                AX7    17                                              SQZ         2095     A
SQZ                NZ     X7,SRCH     IF SHIFT(VAL,-17) " 0               SQZ         2096          I
          -CCG17
SQZ                NZ     X7,CCV3     IF VAL " SMALL CONSTANT             CCG17        476     A
SQZ                S"OC"  OC.S        OC = OC.S                           SQZ         2097     A
SQZ                MX3    42                                              CCG17        477     A
SQZ                BX6    -X3*X6                                          CCG17        478     A
SQZ                                                                       SQZ         2098     A
SQZ        CCV2    LX6    R1.INP                                          SQZ         2099     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                RJ     URI         URI( OC  VAL,RI )                          SQZ        2100     A
SQZ                MX7    0                                                      SQZ        2101     A
SQZ                SA7    A5+B1       [R1+1] = 0                                 SQZ        2102     A
SQZ                EQ     FMA                                                    SQZ        2103     A
SQZ                                                                              CCG17       479     A
SQZ       CCV3     SB4    "OC"-OC.LDC                                            CCG17       480     A
SQZ                ZR     B4,LDC1     IF OC = OC.LDC  */ CALLED BY *LDC*         CCG17       481     A
SQZ                                                                              CCG17       482     A
SQZ                S"OC"  OC.LDC      OC = OC.LDC                                CCG17       483     A
SQZ                SA6    A5+B1       [R2] = VAL                                 CCG17       484     A
SQZ                SX6    B1                                                     CCG17       485         I
          -CCG0124
SQZ                SX6    B1+B1                                                  CCG0124       6     A
SQZ                LX6    R1.INP      IN[R1] = 2    */ LD ONLY *UDT* INDEX       CCG17       486     A
SQZ                RJ     URI         URI( OC.LDC  RI,VAL )                      CCG17       487     A
SQZ                SA3    A5+B1                                                  CCG17       488     A
SQZ                BX6    X3          VAL = [R2]                                 CCG17       489     A
SQZ                EQ     LDC1                                                   CCG17       490     A
SQZ       BTERM    EJECT                                                          SQZ        2104         I
          -CCG17
SQZ       BTERM    TITLE  PSEUDO AND JUMP INSTRUCTION PROCESSING                 CCG17       491     A
SQZ      *         BOS - BEGINNING OF STMT MARKER                                 SQZ        2105     A
SQZ                                                                              SQZ        2106     A
SQZ                PROCESS BOS                                                    SQZ        2107     A
SQZ                LX5    -R1.INP                                                 SQZ        2108     A
SQZ                SX6    X5                                                      SQZ        2109     A
SQZ                SA6    =XLINENR    LINENR = IN[R1]  */ LINE NO OF THIS STMT    SQZ        2110         I
          -CCG1
SQZ                SA6    =XCO.CSN     CO.CSN  = IN[R1]  */ LINE NO OF THIS STMT  CCG1         280         I
          -CCGNC
SQZ                SA6    =XHO$CSN     HO$CSN  = IN[R1]  */ LINE NO OF THIS STMT  CCGNC        762     A
SQZ                                                                              SQZ        2111     A
SQZ                PROCESS NOP                                                    SQZ        2112     A
SQZ                MX7    0                                                      SQZ        2113     A
SQZ                SA7    A4          [DI] = 0                                   SQZ        2114     A
SQZ                EQ     MSL1                                                   SQZ        2115     A
SQZ       RS       SPACE  3                                                      SQZ        2116         I
          -FCC0588
SQZ                                                                              FCC0588       5     A
SQZ                PROCESS EOS                                                    FCC0588       6     A
SQZ                SX3    X5                                                      FCC0588       7     A
SQZ                ZR     X3,MSL1     IF IH[R1] = 0 */ NOT A OPT=2 *EOS*         FCC0588       8     A
SQZ                SA2    A5-B2       DIP = R1 - 2                               FCC0588       9     A
SQZ                LX2    59-D.BMP                                               FCC0588      10     A
SQZ                MI     X2,MSL1     IF BM[DIP]    */ LAST INST IS A *BM*       FCC0588      11     A
SQZ                EQ     .NOP                                                   FCC0588      12     A
SQZ                                                                              FCC0588      13     A
SQZ      *         RS - REGISTER STORE                                            SQZ        2117     A
SQZ                                                                              SQZ        2118     A
SQZ                PROCESS RS                                                     SQZ        2119     A
SQZ                ZR     X4,MSL1     IF [DI] = 0   */ DEAD INSTRUCTION          SQZ        2120     A
SQZ                                                                              SQZ        2121     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                 SA1    "RN"+B6     RI = RND(RI)                                  SQZ          2122     A
SQZ                 BX6    X0*X5                                                     SQZ          2123     A
SQZ                 BX2    -X0*X1                                                    SQZ          2124     A
SQZ                 IX7    X6+X2                                                     SQZ          2125     A
SQZ                 SA7    A5                                                        SQZ          2126     A
SQZ                 BX6    X7-X5                                                     SQZ          2127     A
SQZ                 BX5    X7                                                        SQZ          2128     A
SQZ                 LX7    59-R1.INP                                                 SQZ          2129     A
SQZ                 MI     X7,MSL1     IF IN[R1] " 0  */ RJ *RS*                     SQZ          2130     A
SQZ                                                                                  SQZ          2131     A
SQZ                 LX1    -RN.TPP                                                   SQZ          2132     A
SQZ                 SA2    X1+3                                                      SQZ          2133     A
SQZ                 SX3    A5                                                        SQZ          2134     A
SQZ                 LX3    36                                                        SQZ          2135          I
          -CCG17
SQZ                 LX3    I.RSP                                                     CCG17         492     A
SQZ                 BX7    X3+X2       RSL[TP[RI]+3] = R1                            SQZ          2136     A
SQZ                 SA3    RSS                                                       CCG17         493     A
SQZ                 SA7    A2                          */ LINK DEF TO *RS*           SQZ          2137     A
SQZ                 ZR     X6,RS1      IF RI DIDN"T CHANGE                           SQZ          2138     A
SQZ                                                                                  SQZ          2139     A
SQZ                 SA2    RSS                                                       SQZ          2140          I
          -CCG17
SQZ                 BX6    X2                                                        SQZ          2141          I
          -CCG17
SQZ                 SA6    A4+B1       [DI+1] = RSS ( ADD TO CHAIN OF SEPARATED      SQZ          2142          I
          -CCG17
SQZ                 SX7    A4+B1                       REGISTER STORES )             SQZ          2143          I
          -CCG17
SQZ                 SA7    A2          RSS = DI + 1                                  SQZ          2144          I
          -CCG17
SQZ                 BX6    X3                                                        CCG17         494     A
SQZ                 SA6    A4+B1       [DI+1] = RSS  */                              CCG17         495     A
SQZ                 SB5    "TXT"                                                     CCG17         496     A
SQZ                 SX7    A5-B5                                                     CCG17         497     A
SQZ                 SA7    A3          RSS = R1 - TXT */                             CCG17         498     A
SQZ                                                                                  SQZ          2145     A
SQZ       *         BUMP USES COUNT OF OPERAND IF *UJP* OR FULL LOCK *RS*.          SQZ          2146     A
SQZ                                                                                  SQZ          2147     A
SQZ        RS1      BX6    X5                                                        SQZ          2148     A
SQZ                 LX6    59-R1.SOP-SO.LKP                                          SQZ          2149     A
SQZ                 MI     X6,MSL1     IF IF SOTLK[R1]   */ DONT COLLECT USES        SQZ          2150     A
SQZ                 LX5    R1.RJP                                                    SQZ          2151     A
SQZ                 LX6    -1                                                        CCGA083        15     A
SQZ                 MI     X6,AUC.RJ   IF FULL LOCK                                  CCGA083        16     A
SQZ                 SA2    FAC                                                       CCGA083        17     A
SQZ                 SA3    A4+B1                                                     CCGA083        18     A
SQZ                 LX2    18                                                        CCGA083        19     A
SQZ                 BX7    X2+X3       AC[LI] = FAC                                  CCGA083        20     A
SQZ                 SA7    A3                                                        CCGA083        21     A
SQZ                 SX6    A4+B1                                                     CCGA083        22     A
SQZ                 SA6    A2          FAC = DI+1                                    CCGA083        23     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                EQ     AUC.RJ                                                  SQZ        2152     A
SQZ      RJ3       SPACE  3,14                                                    CCGA083      24     A
SQZ      *         RJ3 - ELIMINATE REDUNDANT BEF CALLS                            CCGA083      25     A
SQZ                                                                               CCGA083      26     A
SQZ                PROCESS RJ3                                                    CCGA083      27     A
SQZ       .F       IFEQ   HC.ID,2     FTN ONLY                                    CCGA083      28     A
SQZ                SX6    1                                                       CCGA083      29     A
SQZ                SA6    CC$XR       CC$XR = 1                                   CCGA083      30     A
SQZ                MX7    0                                                       CCG0211       6     A
SQZ                SA7    RJCA                                                    CCG0211       7     A
SQZ                SA3    RJ3C                                                    CCGA083      31     A
SQZ                SA7    A7+1                                                    CCG0211       8     A
SQZ       RJ1      ZR     X3,RJ7      IF I = 0  */ END OF CHAIN                   CCGA083      32     A
SQZ                SA2    X3+         R1 = [I]                                    CCGA083      33     A
SQZ                SA3    X3+3        LI = [I+3]                                  CCGA083      34     A
SQZ                BX7    X3                                                      CCGA083      35     A
SQZ                SX3    X3          I = RJ3C[LI]                                CCGA083      36     A
SQZ                SA1    A2+4        R1 = R1 + 4                                 CCG0206       6     A
SQZ                UX6    B3,X1                                                   CCG0206       7     A
SQZ                SB4    OC.DEF                                                  CCG0206       8     A
SQZ                NE     B3,B4,RJ1                */ IF NO OC.DEF AFTER RJ3      CCG0206       9     A
SQZ                BX6    X2-X5                                                   CCGA083      37     A
SQZ                NZ     X6,RJ1      IF IH'S .NE.                                CCGA083      38     A
SQZ                                                                               CCGA083      39     A
SQZ                SX6    A3          RJ3T = LI     */ SAVE PTR TO LINK WORD      CCGA083      40     A
SQZ                SA6    RJ3T                                                    CCGA083      41     A
SQZ                SB4    A2          (B4) = SRJ */ R1 OF FIRST RJ                CCGA083      42          I
          -CCG0158
SQZ                                                                               CCGA083      43     A
SQZ      *         IH'S MATCH, SEE IF ARG LISTS MATCH                             CCGA083      44     A
SQZ                                                                               CCGA083      45     A
SQZ                AX7    36          FACI = AC[LI]                               CCGA083      46     A
SQZ                SA2    FAC                                                     CCGA083      47     A
SQZ                SB6    B1          SI = 1                                      CCGA083      48     A
SQZ                MX6    0                                                       CCGA083      49     A
SQZ                SA6    STK         STK(0) = 0                                  CCGA083      50     A
SQZ                                                                               CCGA083      51     A
SQZ      *         X2,X7 = PTRS TO LINK WDS OF RS'S (THIS,PREVIOUS)               CCGA083      52     A
SQZ                                                                               CCGA083      53     A
SQZ       RJ2      SA3    X2                                                      CCGA083      54     A
SQZ                AX3    18                                                      CCGA083      55     A
SQZ                SX6    X3          LWA = AC[LWA]                               CCGA083      56     A
SQZ                SA3    X7                                                      CCGA083      57     A
SQZ                AX3    18                                                      CCGA083      58     A
SQZ                SX3    X3          LWB = AC[LWB]                               CCGA083      59     A
SQZ                LX6    18                                                      CCGA083      60     A
SQZ                BX6    X6+X3                                                   CCGA083      61     A
SQZ                SA6    FACS        FACS = LW(0,0,LWA,LWB)                      CCGA083      62     A
SQZ                SA1    X2-3                                                    CCGA083      63     A
SQZ                BX1    -X0*X1      RIA = RI[LWB-3]                             CCGA083      64     A
SQZ                SA2    X7-3                                                    CCGA083      65     A
SQZ                BX2    -X0*X2      RIB = RI[LWB-3]                             CCGA083      66     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                                                                                  CCGA083     67    A
SQZ      RJ3     ZR     X1,RJ8      IF RIA = 0                                      CCGA083     68    A
SQZ              ZR     X2,RJ8      IF RIB = 0                                      CCGA083     69    A
SQZ              SA1    B7+X1                                                       CCGA083     70    A
SQZ              LX1    -RN.TPP                                                     CCGA083     71    A
SQZ              SA1    X1          R1A = TP[RND(RIA)]                              CCGA083     72    A
SQZ              SA2    B7+X2                                                       CCGA083     73    A
SQZ              LX2    -RN.TPP                                                     CCGA083     74    A
SQZ              SA2    X2          R1B = TP[RND(RIB)]                              CCGA083     75    A
SQZ              BX6    X1-X2                                                       CCGA083     76    A
SQZ              SA3    A1+B2                                                       CCGA083     77    A
SQZ              MX7    D.TYL                                                       CCGA083     78    A
SQZ              LX3    58-D.TYP                                                    CCGA083     79    A
SQZ              BX7    X7*X3                                                       CCGA083     80    A
SQZ              ZR     X7,RJ5      IF TY[R1A+1] = I                                CCGA083     81          I
         -CCG0147
SQZ              ZR     X7,RJ5      IF TY[R1A+2] = I                                CCG0147      5    A
SQZ              AX6    R1.INP                                                      CCGA083     82    A
SQZ              NZ     X6,RJ6      IF (OC,IN)[R1A] .NE. (OC,IN)[R1B]               CCGA083     83    A
SQZ              MI     X3,RJ3A     IF TY[R1A+2] = III                              CCGA083     84    A
SQZ              UX6    B3,X1                                                       CCGA083     85    A
SQZ              SX6    B3-OC.DEF                                                   CCGA083     86    A
SQZ              ZR     X6,RJ6      IF OC = OC.DEF  */ NESTED FUNCTION CALLS        CCGA083     87    A
SQZ              EQ     RJ4                                                         CCGA083     88    A
SQZ      RJ3A    RJ     MMR         MATCH MEM REFS                                  CCGA083     89    A
SQZ              NZ     X3,RJ3      IF INDEXED REFS   */ FOLLOW PRED                CCGA083     90    A
SQZ                                                                                  CCGA083     91    A
SQZ      *       POP STACK                                                           CCGA083     92    A
SQZ                                                                                  CCGA083     93    A
SQZ      RJ4     SB6    B6-B1       SI = SI - 1                                     CCGA083     94    A
SQZ              SA2    STK+B6                                                      CCGA083     95    A
SQZ              SX1    X2                                                          CCGA083     96    A
SQZ              AX2    18          RIA,RIB = STK(I)                                CCGA083     97    A
SQZ              EQ     RJ3                                                         CCGA083     98    A
SQZ                                                                                  CCGA083     99    A
SQZ      *       TYPE I - PUSH STACK                                                 CCGA083    100    A
SQZ                                                                                  CCGA083    101    A
SQZ      RJ5     AX6    R1.OCP                                                      CCGA083    102    A
SQZ              NZ     X6,RJ6      IF OC[R1A] .NE. OC[R1B]                         CCGA083    103    A
SQZ              LX3    1+D.TYP-D.LDP                                               CCG0147      6    A
SQZ              MI     X3,RJ3A     IF LD[DI]                                       CCG0147      7    A
SQZ              AX1    R1.RIL                                                      CCGA083    104    A
SQZ              AX2    R1.RIL                                                      CCGA083    105    A
SQZ              BX3    -X0*X1                                                      CCGA083    106    A
SQZ              BX6    -X0*X2                                                      CCGA083    107    A
SQZ              LX6    18                                                          CCGA083    108    A
SQZ              BX6    X6+X3                                                       CCGA083    109    A
SQZ              SA6    STK+B6      STK(SI) = STKW(RKA,RKB)                         CCGA083    110    A
SQZ              SB6    B6+B1       SI = SI + 1                                     CCGA083    111    A
SQZ              AX1    R1.RIL                                                      CCGA083    112    A
SQZ              AX2    R1.RIL                                                      CCGA083    113    A
SQZ              BX1    -X0*X1      RIA = RJ[R1A]                                   CCGA083    114    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ              BX2    -X0*X2      RIB = RJ[R1B]                                  CCGA083     115     A
SQZ              NZ     X1,RJ3      IF RIA .NE. 0                                  CCGA083     116     A
SQZ              NZ     X2,RJ6      IF RIB .NE. 0                                  CCGA083     117     A
SQZ              EQ     RJ4         GO POP THE OTHER OPERAND                       CCGA083     118     A
SQZ                                                                               CCGA083     119     A
SQZ      *       FAIL EXIT, TRY AGAIN                                             CCGA083     120     A
SQZ                                                                               CCGA083     121     A
SQZ       RJ6    SA3    RJ3T                                                      CCGA083     122     A
SQZ              SA3    X3                                                        CCGA083     123     A
SQZ              SX3    X3          I = SETX([RJ3T])  */ RESTORE LINK TO NEXT     CCGA083     124     A
SQZ              EQ     RJ1                                                       CCGA083     125     A
SQZ                                                                               CCGA083     126     A
SQZ      *       FAIL, NO MATCH OF CURRENT ENTRY ON RJ3 CHAIN                     CCGA083     127     A
SQZ                                                                               CCGA083     128     A
SQZ       RJ7    SA1    FAC                                                       CCGA083     129     A
SQZ              SA2    MRC                                                       CCGA083     130     A
SQZ              SA3    RJ3C                                                      CCGA083     131     A
SQZ              LX1    36                                                        CCGA083     132     A
SQZ              LX2    18                                                        CCGA083     133     A
SQZ              BX6    X1+X2                                                     CCGA083     134     A
SQZ              BX6    X6+X3                                                     CCGA083     135     A
SQZ              SA6    A4+B1       [LI] = LW(0,FAC,MRC,RJ3C)                     CCGA083     136     A
SQZ              CLEAR              CLEAR CHAIN POINTERS                          CCGA083     137     A
SQZ              SX7    A5                                                        CCGA083     138     A
SQZ              SA7    A3          RJ3C = R1                                     CCGA083     139     A
SQZ              EQ     MSL1                                                      CCGA083     140     A
SQZ                                                                               CCGA083     141     A
SQZ      *       ONE OR BOTH RI ARE ZERO                                          CCGA083     142     A
SQZ                                                                               CCGA083     143     A
SQZ       RJ8    BX6    X1-X2                                                     CCGA083     144     A
SQZ              NZ     X6,RJ6      IF NOT(RIA = 0 AND RIB = 0)                   CCGA083     145     A
SQZ              GT     B6,B1,RJ4   IF SI > 1     */ STACK NON-EMPTY              CCGA083     146     A
SQZ                                                                               CCGA083     147     A
SQZ      *       SUCCESS                                                          CCGA083     148     A
SQZ                                                                               CCGA083     149     A
SQZ              SA3    FACS                                                      CCGA083     150     A
SQZ              ZR     X3,RJ9      IF FACS = 0   */ NO MORE ARGS                 CCGA083     151     A
SQZ              SX2    X3                                                        CCGA083     152         I
         -CCG0189
SQZ              SX7    X3                                                        CCG0189       6     A
SQZ              AX3    18                                                        CCGA083     153     A
SQZ              SB6    B1          SI = 1                                        CCGA083     154     A
SQZ              SX7    X3                                                        CCGA083     155         I
         -CCG0189
SQZ              SX2    X3                                                        CCG0189       7     A
SQZ              EQ     RJ2         GO CHECK NEXT ARG                             CCGA083     156     A
SQZ                                                                               CCGA083     157     A
SQZ      *       ELIMINATE THIS RJ, STORE RESULT OF PREVIOUS IF NECESSARY         CCGA083     158     A
SQZ                                                                               CCGA083     159     A
SQZ       RJ9    SA1    FAC         I = FAC                                       CCGA083     160     A
SQZ              SX5    B1                                                        CCGA083     161     A
SQZ              LX5    RN.USESP                                                  CCGA083     162     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ              MX7    0            RSSV = 0                                    CCGA083    163    A
SQZ              SA7    A1           FAC = 0                                     CCGA083    164    A
SQZ       RJ10   SA2    X1           LI = I                                      CCGA083    165    A
SQZ              MX6    60                                                       CCGA083    166    A
SQZ              SA6    A2-B1        [LI-1] = -0  */ CLEAR DESCR OF RS           CCGA083    167    A
SQZ              SA3    A6-B2        R1 = LI - 3                                 CCGA083    168    A
SQZ              BX3    -X0*X3                                                   CCGA083    169    A
SQZ              SA3    B7+X3                                                    CCGA083    170    A
SQZ              IX6    X3-X5        USES[RND(RI)] = USES[RND(RI)] - 1           CCGA083    171    A
SQZ              SA6    A3                                                       CCGA083    172    A
SQZ              LX6    -RN.TPP                                                  CCGA083    173    A
SQZ              SA3    X6+B2        DIP = TP[RND(RI)] + 2                       CCGA083    174    A
SQZ              SX6    B1                                                       CCGA083    175    A
SQZ              LX6    D.PRSP                                                   CCGA083    176    A
SQZ              BX6    -X6*X3                                                   CCGA083    177    A
SQZ              SA6    A3           PRS[DIP] = 0                                CCGA083    178    A
SQZ              SX3    X2+          RSC = RSS[LI] */ SEPERATED RS CHAIN         CCGA083    179    A
SQZ              ZR     X3,RJ11      IF RSC = 0                                  CCGA083    180    A
SQZ              SX7    X3+          RSSV = RSC                                  CCGA083    181    A
SQZ       RJ11   LX2    -18                                                      CCGA083    182    A
SQZ              SX1    X2           I = AC[LI]                                  CCGA083    183    A
SQZ              NZ     X1,RJ10      IF I .NE. 0                                 CCGA083    184    A
SQZ              ZR     X7,RJ12      IF RSSV = 0                                 CCGA083    185    A
SQZ              SA7    RSS          RSS = RSSV    */ UPDATE RS CHAIN            CCGA083    186    A
SQZ                                                                              CCGA083    187    A
SQZ       RJ12   SA3    RJ3T                                                     CCGA083    188    A
SQZ              SB5    X3+B1        R1D = RJ3T+1                                CCGA083    189    A
SQZ              SB6    B1           PREC = 1                                    CCGA083    190    A
SQZ              MX7    59                                                       CCGA083    191    A
SQZ              SA7    CAS          CAS = -1                                    CCGA083    192    A
SQZ              SA7    A7+B1                                                    CCGA083    193    A
SQZ              SA3    B5+4         R1S = R1D+4   */ SKIP OVER FIRST DEF        CCGA083    194    A
SQZ              SB4    OC.ST                                                    CCGA083    195    A
SQZ              UX6    B3,X3                                                    CCGA083    196    A
SQZ              SA2    B5+B1        R2D = R1D+1                                 CCGA083    197    A
SQZ              ZR     X2,RJ13      IF [R2D] = 0                                CCGA083    198    A
SQZ                                                                              CCGA083    199    A
SQZ              LX2    -IH.CAP                                                  CCGA083    200    A
SQZ              SX6    X2                                                       CCGA083    201    A
SQZ              SA6    CAS          CAS = CA[R2D] */ CA FROM PREVIOUS SQZ       CCGA083    202    A
SQZ              NE     B3,B4,RJ12A  IF OC[R1S] .NE. OC.ST                       CCGA083    203    A
SQZ              SA3    A3+4         R1S = R1S + 4                               CCGA083    204    A
SQZ       RJ12A  UX6    B3,X3                                                    CCGA083    205    A
SQZ              SB4    OC.DEF                                                   CCGA083    206    A
SQZ              SB5    A2           FIRST = R2D                                 CCGA083    207    A
SQZ              SX7    4                                                        CCGA083    208    A
SQZ              NE     B3,B4,RJ13A   IF OC[R1S] .NE. OC.DEF  */ NOT DBL PREC    CCGA083    209    A
SQZ              SB6    B1+B1        PREC = 2                                    CCGA083    210    A
SQZ              SA3    A3+B1                                                    CCGA083    211    A
SQZ              LX3    -IH.CAP                                                  CCGA083    212    A
SQZ              SX6    X3           CAS(2) = CA[R1S+1]                          CCGA083    213    A
SQZ              SA6    CAS+1                                                    CCGA083    214    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                EQ     RJ13A                                                    CCGA083     215     A
SQZ                                                                                CCGA083     216     A
SQZ       RJ13     NE     B3,B4,RJ15  IF OC[R1S] .NE. OC.ST                        CCGA083     217     A
SQZ                SA1    =XS=ST                                                   CCGA083     218     A
SQZ                SA2    A3+B1                                                    CCGA083     219     A
SQZ                BX6    X1-X2                                                    CCGA083     220     A
SQZ                SX7    X6+                                                      CCGA083     221     A
SQZ                NZ     X7,RJ14     IF IH[R1S+1] .NE. ST.                        CCGA083     222     A
SQZ                LX2    -IH.CAP                                                  CCGA083     223     A
SQZ                SX6    X2                                                       CCGA083     224     A
SQZ                SA6    CAS         CAS = CA[R1S+1]                              CCGA083     225     A
SQZ                SA3    A3+4        R1S = R1S+4                                  CCGA083     226     A
SQZ                UX6    B3,X3                                                    CCGA083     227     A
SQZ                SB5    A2          FIRST = R2S                                  CCGA083     228     A
SQZ                MX3    R1.CAL                                                   CCG0211       9     A
SQZ                SA2    A2-1                        */ R1W                       CCG0211      10     A
SQZ                LX3    18+R1.CAP                                                CCG0211      11     A
SQZ                BX6    X3*X2                        */ LINK TO UDI              CCG0211      12     A
SQZ                SA6    RJCA                          */ USED IN GTL             CCG0211      13     A
SQZ                SX7    8                                                        CCGA083     229     A
SQZ                SB4    OC.DEF                                                   CCGA083     230     A
SQZ                NE     B3,B4,RJ13A  IF OC[R1S] .NE. OC.DEF  */ NOT DBL PRECISI  CCGA083     231     A
SQZ                SA2    A3+5                                                     CCGA083     232     A
SQZ                LX2    -IH.CAP                                                  CCGA083     233     A
SQZ                SX6    X2                                                       CCGA083     234     A
SQZ                SA6    CAS+1       CAS(2) = CA[R1S+5]                           CCGA083     235     A
SQZ                SB6    B1+B1       PREC= 2                                      CCGA083     236     A
SQZ                SA2    A2-1                            */ R1W                   CCG0211      14     A
SQZ                BX6    X3*X2                                                    CCG0211      15     A
SQZ                SA6    RJCA+1                           */ GTL                  CCG0211      16     A
SQZ                                                                               CCGA083     237     A
SQZ       *        RESULT OF RJ3 IS STORED TO AN ST. , ENSURE THAT THERE IS NO      CCGA083     238     A
SQZ       *        REDEFINITION OF THE ST. BETWEEN THE CURRENT AND THE FIRST RJ3.   CCGA083     239     A
SQZ       *        B5 = LOC OF R2 OF FIRST ST., X7 = INCREMENT BETWEEN ST.'S        CCGA083     240     A
SQZ                                                                               CCGA083     241     A
SQZ        RJ13A   SA7    RJ3T        SAVE INCREMENT IN CASE DBL PREC              CCGA083     242     A
SQZ                SB3    B5          B3 = FIRST                                   CCGA083     243     A
SQZ                MX7    0           PASS = 0                                     CCGA083     244     A
SQZ                SA3    B3+B1                                                    CCGA083     245     A
SQZ                NZ     X3,RJ13B    IF [DI] .NE. 0  */ STORE IS LIVE             CCGA083     246     A
SQZ                SA3    B3-B1                                                    CCGA083     247     A
SQZ                UX3    B4,X3                                                    CCGA083     248     A
SQZ                BX3    -X0*X3                                                   CCGA083     249     A
SQZ                SA3    B7+X3       RND(RI[R1])                                  CCGA083     250     A
SQZ                SX2    B1                                                       CCGA083     251     A
SQZ                LX2    RN.USESP                                                 CCGA083     252     A
SQZ                IX6    X3+X2       BUMP USES CNT OF ST PRED                     CCGA083     253     A
SQZ                SA6    A3                                                       CCGA083     254     A
SQZ                SA3    F.RDT+B4                                                 CCGA083     255     A
SQZ                BX6    X3                                                       CCGA083     256     A
SQZ                SA6    B3+B1       RESTORE DESCRIPTOR                           CCGA083     257     A
SQZ                                                                               CCGA083     258     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ       RJ13B    MX2    -IH.CAIHL                                          CCGA083     259     A
SQZ                SA3    B3          R2W = [FIRST] */ R2W OF ST TO ST.      CCGA083     260     A
SQZ                BX4    -X2*X3      PAT = CAIH[R2W]                        CCGA083     261     A
SQZ                SB5    B0          LOC = 0  */ NO INTERFERENCE            CCGA083     262     A
SQZ                SA3    A4-1-4      R2I = DI-1-4                           CCGA083     263     A
SQZ                                                                         CCGA083     264     A
SQZ      *         SEARCH BACKWARD FOR REDEF OF THIS ST.                     CCGA083     265     A
SQZ                                                                         CCGA083     266     A
SQZ       RJ13C    SA3    A3-4        R2I = R2I - 4                          CCGA083     267     A
SQZ                BX5    X3-X4                                              CCGA083     268     A
SQZ                BX6    -X2*X5                                             CCGA083     269     A
SQZ                NZ     X6,RJ13C    IF CAIH[R2I] .NE. PAT                  CCGA083     270     A
SQZ                                                                         CCGA083     271     A
SQZ                SB4    A3                                                 CCGA083     272     A
SQZ                LE     B4,B3,RJ13D IF R2I = FIRST  */ IF FINISHED         CCGA083     273     A
SQZ                SA1    A3-B1                                              CCGA083     274     A
SQZ                UX6    B4,X1                                              CCGA083     275     A
SQZ                SX6    B4-OC.ST                                           CCGA083     276     A
SQZ                NZ     X6,RJ13C    IF OC[R2I-1] .NE. OC.ST  */ NOT A REDEF CCGA083     277     A
SQZ                SA1    A3+B1                                              CCGA083     278     A
SQZ                ZR     X1,RJ13C    IF DESCR = 0  */ INSTRUCTION IS DEAD   CCGA083     279     A
SQZ                                                                         CCGA083     280     A
SQZ                SB5    A3          LOC = R2I                              CCGA083     281     A
SQZ                EQ     RJ13C                                              CCGA083     282     A
SQZ                                                                         CCGA083     283     A
SQZ       RJ13D    SX7    X7+B1       PASS = PASS + 1                        CCGA083     284     A
SQZ                ZR     B5,RJ13X    IF LOC = 0  */ NO REDEFINITIONS        CCGA083     285     A
SQZ                                                                         CCGA083     286     A
SQZ      *         ST.+C IS REDEFINED, PICK A NEW CA AND UPDATE REFS         CCGA083     287     A
SQZ                                                                         CCGA083     288     A
SQZ                MX6    0                                                  CCG0211      17     A
SQZ                SA6    RJCA                                               CCG0211      18     A
SQZ                SA6    RJCA+1                                             CCG0211      19     A
SQZ                SA1    STBASE      N = STBASE                             CCGA083     289     A
SQZ                BX6    X1                                                 CCGA083     290     A
SQZ                SA6    CAS-1+X7    CAS(PASS) = N                          CCGA083     291     A
SQZ                SX6    X1+B1       STBASE += 1                            CCGA083     292     A
SQZ                SA6    A1                                                 CCGA083     293     A
SQZ                SA5    S=ST                                               CCGA083     294     A
SQZ                LX1    IH.CAP                                             CCGA083     295     A
SQZ                BX1    X1+X5       NEWPAT = CAIH(N,ST.)                   CCGA083     296     A
SQZ                SA3    B5          R2J = LOC                              CCGA083     297     A
SQZ                                                                         CCGA083     298     A
SQZ       RJ13E1   MX2    -IH.CAIHL                                          CCGA094       5     A
SQZ                                                                         CCGA094       6     A
SQZ       RJ13E    SA3    A3-4        R2J = R2J - 4                          CCGA083     299     A
SQZ                BX5    X3-X4                                              CCGA083     300     A
SQZ                BX6    -X2*X5                                             CCGA083     301     A
SQZ                NZ     X6,RJ13E    IF CAIH[R2J] .NE. PAT                  CCGA083     302     A
SQZ                SA5    A3+B1                                              CCG0181       6           I
          -CCGA094
SQZ                ZR     X5,RJ13E    IF DESCR = 0                           CCG0181       7           I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
-CCGA094
SQZ                SA2    A3+B1                                    CCGA094        7     A
SQZ                ZR     X2,RJ13E1   IF DESCR = 0                 CCGA094        8     A
SQZ                                                                CCGA083      303     A
SQZ                MX2    -IH.CAIHL                                CCGA094        9     A
SQZ                BX3    X2*X3                                    CCGA083      304     A
SQZ                BX6    X3+X1       CAIH[R2J] = NEWPAT           CCGA083      305     A
SQZ                SA6    A3                                       CCGA083      306     A
SQZ                SB4    A3                                       CCGA083      307     A
SQZ                SX2    A0-B4                                    CCGA083      308           I
-CCG0165
SQZ                BX2    -X2                                      CCGA083      309           I
-CCG0165
SQZ                GT     B4,B3,RJ13E IF R2J GT FIRST              CCGA083      310     A
SQZ                                                                CCGA083      311     A
SQZ       RJ13X    SB4    X7                                       CCGA083      312     A
SQZ                EQ     B4,B6,RJ18  IF PASS = PREC               CCGA083      313     A
SQZ                                                                CCGA083      314     A
SQZ       *        SETUP FOR SECOND HALF OF DOUBLE PREC             CCGA083      315     A
SQZ                                                                CCGA083      316     A
SQZ                SA1    RJ3T                                     CCGA083      317     A
SQZ                SB3    B3+X1                                    CCGA083      318     A
SQZ                EQ     RJ13B                                    CCGA083      319     A
SQZ                                                                CCGA083      320     A
SQZ                                                                CCGA083      321     A
SQZ       RJ14     SA3    A3+4        R1S = R1S+4                  CCGA083      322     A
SQZ                UX6    B3,X3                                    CCGA083      323     A
SQZ       RJ15     SB4    OC.DEF                                   CCGA083      324     A
SQZ                NE     B3,B4,RJ16                               CCGA083      325     A
SQZ                SB6    B1+B1       PREC = 2                     CCGA083      326     A
SQZ       RJ16     SA1    STBASE                                   CCGA083      327     A
SQZ                BX6    X1                                       CCGA083      328     A
SQZ                SA6    CAS         CAS = STBASE                 CCGA083      329     A
SQZ                SX6    X6+B1                                    CCGA083      330     A
SQZ                SA6    A6+B1       CAS(2) = STBASE+1            CCGA083      331     A
SQZ                SX7    X1+B6       STBASE = STBASE + PREC       CCGA083      332     A
SQZ                SA7    A1                                       CCGA083      333     A
SQZ                SA2    S=ST                                     CCGA083      334     A
SQZ                LX1    IH.CAP                                   CCGA083      335     A
SQZ                SX3    B1                                       CCGA083      336     A
SQZ                LX3    58                                       CCGA083      337     A
SQZ                BX6    X1+X2                                    CCGA083      338     A
SQZ                BX6    X6+X3                                    CCGA083      339     A
SQZ                SA6    IXF         IXF = NON_ZERO               CCGA083      340     A
SQZ                SA6    B5+B1       [R1D+1] = IHW(F,0,CAS,ST.)   CCGA083      341     A
SQZ                RJ     BDU         BUMP USES OF DEF             CCGA083      342     A
SQZ                EQ     B6,B1,RJ18  IF PREC = 1                  CCGA083      343     A
SQZ                SB4    OC.DEF                                   CCGA083      344     A
SQZ                UX7    X6          CLEAR FLAG BIT               CCGA083      345     A
SQZ                SA7    A6                                       CCGA083      346     A
SQZ       RJ17     SB5    B5+4        R1D = R1D+4                  CCGA083      347     A
SQZ                SA3    B5                                       CCGA083      348     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ              UX7    B3,X3                                                  CCGA083     349     A
SQZ              NE     B3,B4,RJ17  IF OC[R1D] .NE. OC.DEF                     CCGA083     350     A
SQZ              SX3    B1                                                     CCGA083     351     A
SQZ              LX3    IH.CAP                                                 CCGA083     352     A
SQZ              IX6    X6+X3                                                  CCGA083     353     A
SQZ              LX3    57-IH.CAP                                              CCGA083     354     A
SQZ              IX6    X6+X3                                                  CCGA083     355     A
SQZ              SA6    A3+B1       [R1D+1] = IHW(F,0,CAS+1,ST.)               CCGA083     356     A
SQZ              RJ     BDU         BUMP USES OF DEF                           CCGA083     357     A
SQZ                                                                            CCGA083     358     A
SQZ      *       CHANGE CURRENT RJ3/DEF TO LD/XMT                              CCGA083     359     A
SQZ                                                                            CCGA083     360     A
SQZ       RJ18   SA3    A4+B2       R1D = DI+2                                 CCGA083     361     A
SQZ              BX6    -X0*X3                                                 CCGA083     362         I
         -CCG0193
SQZ              SB3    OC.DEF                                                 CCG0193       6     A
SQZ              UX6    B5,X3                                                  CCG0193       7     A
SQZ              EQ     B3,B5,RJ19                          */ IF OC.DEF       CCG0193       8     A
SQZ              MX7    -0                                                     CCG0193       9     A
SQZ              SA7    A4                                 */RJ3 IS DEAD       CCG0193      10     A
SQZ              EQ     RJ7                                                    CCG0193      11     A
SQZ                                                                            CCG0193      12     A
SQZ       RJ19   BX6    -X0*X3                                                 CCG0193      13     A
SQZ              SB5    X6          R = RI[R1D]                                CCGA083     363     A
SQZ              SX6    B5-B1                                                  CCGA083     364     A
SQZ              SA2    CAS                                                    CCGA083     365     A
SQZ              SB4    A4-B2                                                  CCGA083     366     A
SQZ              SA1    RJCA                                                   CCG0211      20     A
SQZ              BX7    X1                                                     CCG0211      21     A
SQZ              SA7    GTLC                                */ USED IN GTL     CCG0211      22     A
SQZ              RJ     GTL         GTL( R-1 , CAS , DI-2 )                    CCGA083     367     A
SQZ              SX6    B5-B1                                                  CCGA083     368     A
SQZ              SB4    B4+4                                                   CCGA083     369     A
SQZ              RJ     GXI         GENERATE XMT                               CCGA083     370     A
SQZ              SA4    A4-4        DI = DI - 4   */ BACKUP TO PROCESS LD ST.  CCGA083     371     A
SQZ              EQ     B6,B1,RJ20  IF PREC = 1                                CCGA083     372     A
SQZ              SX6    B5-2                                                   CCGA083     373         I
         -CCG0216
SQZ              SA2    MRC                                                    CCG0216       8     A
SQZ              ZR     X2,RJ19A                  */ IF REFERENCE CHAIN ZERO  CCG0216       9     A
SQZ              BX7    -X2                                                    CCG0216      10     A
SQZ              SB3    A4+1                       */ LI = DI + 1              CCG0216      11     A
SQZ              SX1    X7+B3                                                  CCG0216      12     A
SQZ              NZ     X1,RJ19A                  */ IF MRC DOESN'T POINT HERE CCG0216      13     A
SQZ              SA1    B3                                                     CCG0216      14     A
SQZ              SX6    X1                                                     CCG0216      15     A
SQZ              SA6    A2                         */ UPDATE REFERENCE CHAIN   CCG0216      16     A
SQZ                                                                            CCG0216      17     A
SQZ       RJ19A  SX6    B5-2                                                   CCG0216      18     A
SQZ              SA2    CAS+1                                                  CCGA083     374     A
SQZ              SB4    B4-8                                                   CCGA083     375     A
SQZ              SA1    RJCA+1                                                 CCG0211      23     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                BX7    X1                                                    CCG0211      24    A
SQZ                SA7    GTLC                                 */ USED IN GTL   CCG0211      25    A
SQZ                RJ     GTL         GENERATE SECOND LOAD                       CCGA083     376    A
SQZ                SA3    B4+8                                                   CCGA083     377    A
SQZ      +         SA3    A3+4                                                   CCGA083     378    A
SQZ                UX6    B3,X3                                                  CCGA083     379    A
SQZ                SX7    B3-OC.DEF                                              CCGA083     380    A
SQZ                NZ     X7,*-1                                                 CCGA083     381    A
SQZ                SB4    A3                                                     CCGA083     382    A
SQZ                SX6    B5-2        RJ = RI OF LOAD                            CCGA083     383    A
SQZ                BX7    -X0*X3                                                 CCGA083     384    A
SQZ                SB5    X7                                                     CCGA083     385    A
SQZ                RJ     GXI         REPLACE SECOND DEF WITH XMT                CCGA083     386    A
SQZ                SA4    A4-4        DI = DI - 4   */ BACKUP                    CCGA083     387    A
SQZ       RJ20     BSS    0                                                      CCGA083     388    A
SQZ                EQ     MSL1                                                   CCGA083     389    A
SQZ       BDU      SPACE  3,14                                                   CCGA083     390    A
SQZ      **        BDU - BUMP *DEF* USES, (A6) = R2                              CCGA083     391    A
SQZ                                                                              CCGA083     392    A
SQZ       BDU      ROUTINE                                                       CCGA083     393    A
SQZ                SA3    A6-B1       GET R1 WORD                                CCGA083     394    A
SQZ                BX2    -X0*X3                                                 CCGA083     395    A
SQZ                SA3    B7+X2                                                  CCGA083     396    A
SQZ                SX2    B1                                                     CCGA083     397    A
SQZ                LX2    RN.USESP                                               CCGA083     398    A
SQZ                IX7    X2+X3                                                  CCGA083     399    A
SQZ                SA7    A3                                                     CCGA083     400    A
SQZ                EQ     BDU                                                    CCGA083     401    A
SQZ       GTL      SPACE  3,14                                                   CCGA083     402    A
SQZ      **        GTL - GENERATE TEMP LOAD                                      CCGA083     403    A
SQZ      *         ENTRY X6 = RI                                                 CCGA083     404    A
SQZ      *                X2 = CA                                                CCGA083     405    A
SQZ      *                B4 = R1 ADDR                                           CCGA083     406    A
SQZ                                                                              CCGA083     407    A
SQZ       GTL      ROUTINE                                                       CCGA083     408    A
SQZ                SB3    OC.LD                                                  CCGA083     409    A
SQZ                SX7    B4                                                     CCGA083     410    A
SQZ                LX7    RN.TPP                                                 CCGA083     411    A
SQZ                BX7    X6+X7                                                  CCGA083     412    A
SQZ                SA7    B7+X6       RND(RI) = RNDW(0,0,R1,RI)                  CCGA083     413    A
SQZ                PX6    B3,X6                                                  CCGA083     414    A
SQZ                SA1    GTLC                                                   CCG0211      26    A
SQZ                IX6    X6+X1                     */ INCLUDE LINK TO UDI       CCG0211      27    A
SQZ                SA6    B4                                                     CCGA083     415    A
SQZ                SA1    S=ST                                                   CCGA083     416    A
SQZ                LX2    IH.CAP                                                 CCGA083     417    A
SQZ                BX7    X1+X2                                                  CCGA083     418    A
SQZ                SA7    B4+B1                                                  CCGA083     419    A
SQZ                SA3    =XF$RDT+B3                                             CCGA083     420    A
SQZ                BX6    X3                                                     CCGA083     421    A
SQZ                SA6    A7+B1                                                  CCGA083     422    A
SQZ                MX7    0                                                      CCGA083     423    A

                   0         1         2         3         4         5         6         7         8
                   12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ              SA7    A6+B1       LINK = 0                                  CCGA083     424    A
SQZ              EQ     GTL                                                   CCGA083     425    A
SQZ       GTLC   BSS    1                                                     CCG0211      28    A
SQZ       GXI    SPACE  3,14                                                  CCGA083     426    A
SQZ      **      GXI - GENERATE XMT INSTRUCTION                               CCGA083     427    A
SQZ      *       ENTRY X6 = RJ                                                CCGA083     428    A
SQZ      *              B5  RI                                                CCGA083     429    A
SQZ      *              B4 = R1 ADDR                                          CCGA083     430    A
SQZ                                                                           CCGA083     431    A
SQZ       GXI    ROUTINE                                                      CCGA083     432    A
SQZ              SB3    OC.XMT                                                CCGA083     433    A
SQZ              SA3    F$RDT+B3                                              CCGA083     434    A
SQZ              SA2    B4+B2       DESCR                                     CCGA083     435    A
SQZ              SX7    B1                                                    CCGA083     436    A
SQZ              LX7    D.PSP                                                 CCGA083     437    A
SQZ              BX2    X7*X2                                                 CCGA083     438    A
SQZ              BX3    X2+X3       MOVE TO XMT DESCR                         CCGA083     439    A
SQZ              LX6    R1.RJP                                                CCGA083     440    A
SQZ              SX5    B5                                                    CCGA083     441    A
SQZ              BX6    X6+X5                                                 CCGA083     442    A
SQZ              PX6    B3,X6                                                 CCGA083     443    A
SQZ              SA6    B4                                                    CCGA083     444    A
SQZ              MX7    0                                                     CCGA083     445    A
SQZ              SA7    B4+B1       R2W = 0                                   CCGA083     446    A
SQZ              BX6    X3                                                    CCGA083     447    A
SQZ              SA6    A7+B1                                                 CCGA083     448    A
SQZ              SA7    A6+B1       LINK = 0                                  CCGA083     449    A
SQZ              SX6    B4                                                    CCGA083     450    A
SQZ              LX6    RN.TPP                                                CCGA083     451    A
SQZ              BX6    X5+X6                                                 CCGA083     452    A
SQZ              SX5    B1                                                    CCGA083     453    A
SQZ              LX5    RN.XMTP                                               CCGA083     454    A
SQZ              BX6    X5+X6                                                 CCGA083     455    A
SQZ              SA6    B7+B5       RND(RI) = RNDW(XMT,0,R1,RI)               CCGA083     456    A
SQZ              EQ     GXI                                                   CCGA083     457    A
SQZ       MMR    SPACE  3,14                                                  CCGA083     458    A
SQZ      **      MMR - MATCH MEMORY REFS                                      CCGA083     459    A
SQZ      *       ENTRY A1,A2 = R1 WORD ADDRESS'S OF THE MEM REFS              CCGA083     460    A
SQZ      *       EXIT TO RJ6 IF NO MATCH OR INTERFERENCE                      CCGA083     461    A
SQZ      *       X1,X2,X3 = RF1,RF2,SUM OF RFS                                CCGA083     462    A
SQZ                                                                           CCGA083     463    A
SQZ       MMR    ROUTINE                                                      CCGA083     464    A
SQZ              UX6    B3,X1                                                 CCGA083     465    A
SQZ              SA1    A1+B1       IHA = R1A+1                               CCGA083     466    A
SQZ              SA2    A2+B1       IHB = R1B+1                               CCGA083     467    A
SQZ              BX6    X1-X2                                                 CCGA083     468    A
SQZ              SX3    B3-OC.LDC                                             CCGA083     469    A
SQZ              NZ     X3,MMR1     IF OC[R1A] .NE. OC.LDC                    CCGA083     470    A
SQZ              NZ     X6,RJ6      IF [IHA] .NE. [IHB]                       CCGA083     471    A
SQZ              MI     X6,RJ6      IF COMPLEMENTS                            CCGA083     472    A
SQZ              MX3    0                                                     CCGA083     473    A
SQZ              EQ     MMR                                                   CCGA083     474    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                                                                                  CCGA083     475     A
SQZ       MMR1     MX7     -IH.CAIHL                                                 CCGA083     476     A
SQZ                BX7     -X7*X6                                                    CCGA083     477     A
SQZ                NZ      X7,RJ6      IF NO MATCH OF CA AND IH                      CCGA083     478     A
SQZ                SA6     MMRA                                                      CCGA083     479     A
SQZ                BX7     X2          SAVE DIFF AND RF OF IHB                       CCGA083     480     A
SQZ                LX7     -IH.RFP                                                   CCGA083     481     A
SQZ                SX7     X7                                                        CCGA083     482     A
SQZ                SA7     A6+B1                                                     CCGA083     483     A
SQZ                                                                                  CCGA083     484     A
SQZ       *        SCAN BACKWARD FOR INTERFERING STORE                               CCGA083     485     A
SQZ                                                                                  CCGA083     486     A
SQZ                SB4     A2-B1       B4 = R1B                                      CCG0158       6     A
SQZ                SB4     -B4                                                       CCG0158       7     A
SQZ                SA2     MRC         I = MRC                                       CCGA083     487     A
SQZ                SA3     RJ3C                                                      CCGA083     488     A
SQZ                SB3     X3          J = RJ3C                                      CCGA083     489     A
SQZ                MX5     -IH.IHL                                                   CCGA083     490     A
SQZ       MMR2     ZR      X2,MMR3                                                   CCGA083     491     A
SQZ                SA2     X2          LW = [I]                                      CCGA083     492     A
SQZ                SX2     X2          I = MRC[LW]                                   CCGA083     493     A
SQZ                SX6     A2+B4                                                     CCG0158       8     A
SQZ                MI      X6,MMR4     IF I < R1B                                    CCG0158       9     A
SQZ                SA3     A2-B1       DI = LW - 1                                   CCGA083     494     A
SQZ                LX3     59-D.STP                                                  CCGA083     495     A
SQZ                PL      X3,MMR2     IF NOT ST[DI]                                 CCGA083     496     A
SQZ                SA3     A3-B1       IHW = [DI-1]                                  CCGA083     497     A
SQZ                BX6     X1-X3                                                     CCGA083     498     A
SQZ                BX7     -X5*X6                                                    CCGA083     499     A
SQZ                NZ      X7,MMR2     IF IH[IHW] .NE. IH[R2W]                       CCGA083     500     A
SQZ                LX6     -IH.RFP                                                   CCGA083     501     A
SQZ                BX7     -X5*X6                                                    CCGA083     502     A
SQZ                NZ      X7,RJ6      IF RF'S DIFFERENT                             CCGA083     503     A
SQZ                LX6     IH.RFP-IH.CAP                                             CCGA083     504     A
SQZ                BX7     -X5*X6                                                    CCGA083     505     A
SQZ                ZR      X6,RJ6      IF CA'S THE SAME                              CCGA083     506     A
SQZ                EQ      MMR2                                                      CCGA083     507     A
SQZ                                                                                  CCGA083     508     A
SQZ       MMR3     EQ      B3,B4,MMR4  IF J = SRJ                                    CCGA083     509         I
         -CCG0158
SQZ       MMR3     ZR      B3,MMR4                    */ IF END OF CHAIN             CCG0158      10     A
SQZ                SA3     B3+3        LI = [J+3]                                    CCGA083     510     A
SQZ                SB3     X3          J = RJ3C[LI]                                  CCGA083     511     A
SQZ                LX3     -18                                                       CCGA083     512     A
SQZ                SX2     X3          I = MRC[LI]                                   CCGA083     513     A
SQZ                EQ      MMR2                                                      CCGA083     514     A
SQZ                                                                                  CCGA083     515     A
SQZ       MMR4     LX1     -IH.RFP                                                   CCGA083     516     A
SQZ                SX1     X1                                                        CCGA083     517     A
SQZ                SA2     MMRA+1                                                    CCGA083     518     A
SQZ                BX3     X1+X2                                                     CCGA083     519     A
SQZ                BX6     X1-X2                                                     CCGA083     520     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                ZR     X6,MMR5     IF RF'S =                                    CCGA083      521     A
SQZ                ZR     X3,MMR      IF BOTH = 0                                  CCGA083      522     A
SQZ                EQ     RJ6                                                      CCGA083      523     A
SQZ                                                                                CCGA083      524     A
SQZ       MMR5     SX4    X1-1                                                     CCGA083      525     A
SQZ                NZ     X4,MMR      IF RF .NE. 1                                 CCGA083      526     A
SQZ                MX3    0           INDICATE BOTH ARE 0                          CCGA083      527     A
SQZ                EQ     MMR                                                      CCGA083      528     A
SQZ                                                                                CCGA083      529     A
SQZ       MMRA     BSS    2                                                        CCGA083      530     A
SQZ       RJCA     BSS    2                                                        CCG0211       29     A
SQZ       .F       ENDIF                                                           CCGA083      531     A
SQZ       BTERM    EJECT                                                           SQZ         2153     A
SQZ      *         BLOCK TERMINATORS                                               SQZ         2154     A
SQZ                                                                                SQZ         2155     A
SQZ                PROCESS (UJP,RJ3,RJ6,LAB,ENT)                                   SQZ         2156          I
          -CCG7
SQZ                PROCESS (RJ3,RJ6)                                               CCG7         120          I
          -CCGA083
SQZ                PROCESS RJ6                                                     CCGA083      532     A
SQZ                SX6    1                                                        CCG7         121     A
SQZ                SA6    =XCG.XR     CG.XR = 1     */ INDICATE EXT REFS IN BLOCK  CCG7         122          I
          -CCGNC
SQZ                SA6    =XCC$XR     CC$XR = 1                                    CCGNC        763     A
SQZ                                                                                CCG7         123     A
SQZ                PROCESS (UJP,LAB,ENT)                                           CCG7         124     A
SQZ                CLEAR              CLEAR LINKED LISTS                          SQZ         2157     A
SQZ                EQ     MSL1                                                     SQZ         2158     A
SQZ                                                                                SQZ         2159     A
SQZ                PROCESS JIN                                                     SQZ         2160     A
SQZ                CLEAR              CLEAR LINKED LISTS                          SQZ         2161     A
SQZ                SA1    "RN"+B6                                                  SQZ         2162     A
SQZ                LX1    59-RN.ZP                                                 SQZ         2163          I
          -CCG27
SQZ                PL     X1,.JPX     IF ^Z[RI]                                    SQZ         2164          I
          -CCG27
SQZ                BX6    X0*X5       RI[R1] = 0                                   SQZ         2165          I
          -CCG27
SQZ                SA6    A5                                                       SQZ         2166          I
          -CCG27
SQZ                EQ     MSL1                                                     SQZ         2167          I
          -CCG27
SQZ                BX6    X0*X5                                                    CCG27          7     A
SQZ                BX5    -X0*X1                                                   CCG27          8     A
SQZ                IX6    X6+X5                                                    CCG27          9     A
SQZ                SA6    A5                                                       CCG27         10     A
SQZ                ZR     X5,MSL1     IF RI = 0                                    CCG27         11     A
SQZ                EQ     JPX6                                                     CCG27         12     A
SQZ                                                                                SQZ         2168     A
SQZ      *         CONDITIONAL JUMPS                                               SQZ         2169     A
SQZ                                                                                SQZ         2170     A
SQZ                PROCESS JPX                                                     SQZ         2171     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                SA1    "RN"+B6                                                  SQZ         2172     A
SQZ                BX6    X0*X5                                                    SQZ         2173     A
SQZ                SA2    MRC                                                      SQZ         2174          I
          -CCG27
SQZ                BX5    -X0*X1      RI = RND(RI)                                 SQZ         2175     A
SQZ                LX1    59-RN.XMTCP                                              SQZ         2176     A
SQZ                PL     X1,JPX1     IF ^XMTC[RNI]                                SQZ         2177     A
SQZ                                                                                SQZ         2178     A
SQZ      *         CHANGE *JPX* TO OMIT *XMTC* IF *PL/MI* JUMPS                     SQZ         2179     A
SQZ                                                                                SQZ         2180     A
SQZ                SA3    A5+B1       R2 = R1 + 1                                  SQZ         2181     A
SQZ                LX3    -IH.CAP     JT = CA[R2]   */ JUMP TYPE                   SQZ         2182     A
SQZ                SB5    X3                                                       SQZ         2183     A
SQZ                LT     B5,B2,JPX1  IF JT < 2     */ ZR, NZ                      SQZ         2184     A
SQZ                SB5    B5-B2                                                    SQZ         2185     A
SQZ                GE     B5,B2,JPX1  IF JT > 3     */ RANGE CHECK JUMPS           SQZ         2186     A
SQZ                SX7    B1                                                       SQZ         2187     A
SQZ                BX7    X3-X7                                                    SQZ         2188     A
SQZ                LX7    IH.CAP      JT = XOR(JT,1)   */ TOGGLE *JT*              SQZ         2189     A
SQZ                SA7    A3                                                       SQZ         2190     A
SQZ                LX1    1+RN.XMTCP-RN.TPP                                        SQZ         2191     A
SQZ                SA3    X1          R1P = TP[RNI]                                SQZ         2192     A
SQZ                LX3    -R1.RJP                                                  SQZ         2193     A
SQZ                BX5    -X0*X3      RI = RJ[R1P]  */ SUBSTITUTE OPERAND OF XMTC  SQZ         2194     A
SQZ                SB6    X5                                                       CCGA045       14     A
SQZ                                                                                SQZ         2195     A
SQZ       JPX1     IX6    X6+X5                                                    SQZ         2196     A
SQZ                LX5    R1.RJP                                                   SQZ         2197          I
          -CCG27
SQZ                SA6    A5                                                       SQZ         2198     A
SQZ                SX7    A5          JPXA = R1                                    SQZ         2199          I
          -CCG27
SQZ                SA1    A1                                                       CCG27         13          I
          -CCGA045
SQZ                SA1    "RN"+B6                                                  CCGA045       15     A
SQZ                LX1    59-RN.CP                                                 CCG27         14     A
SQZ                PL     X1,JPX6     IF ^C[RI]                                    CCG27         15     A
SQZ                SB3    OC.UJP                                                   CCG27         16     A
SQZ                RJ     ECJ         EVALUATE CONDITIONAL JUMP                    CCG27         17     A
SQZ                SA2    =XCC$OPF                                                 CCG27         18     A
SQZ                SA3    =XO$CFT                                                  CCG27         19     A
SQZ                LX1    -R1.INP                                                  CCG27         20     A
SQZ                MX7    0                                                        CCG27         21     A
SQZ                PL     B5,JPX2     IF TV \ 0     */ JUMP CONDITION TRUE         CCG27         22     A
SQZ                                                                                CCG27         23     A
SQZ                NZ     X2,MSL1     IF CC$OPF " 0 */ ^OPT=2 & BRIDGE PHASE       CCG27         24     A
SQZ                SX1    X1                                                       CCG27         25     A
SQZ                ZR     X1,MSL1     IF IN[R1] = 0 */ NO FLOW INFO                CCG27         26     A
SQZ                IX1    X1+X3                                                    CCG27         27     A
SQZ                SA7    X1          CFT(IN[R1]) = 0  */ CLEAR FLOW TABLE ENTRY   CCG27         28     A
SQZ                EQ     MSL1                                                     CCG27         29     A
SQZ                                                                                CCG27         30     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ      *         JUMP CONDITION TRUE, MARK REST OF BLOCK AS DEAD                    CCG27        31     A
SQZ                                                                                   CCG27        32     A
SQZ       JPX2     SB4    OC.LAB                                                      CCG27        33     A
SQZ                SB5    OC.JPX                                                      CCG27        34     A
SQZ                SB6    X3                                                          CCG27        35     A
SQZ                SX6    X1          LCI = IN[R1]                                    CCG27        36     A
SQZ                SA3    B6+X6       LCIA = CFT(LCI)                                 CCG27        37     A
SQZ                                                                                   CCG27        38     A
SQZ       JPX3     SA5    A5+4        R1 = R1 + 4                                     CCG27        39     A
SQZ                UX1    "OC",X5                                                     CCG27        40     A
SQZ                ZR     "OC",JPX5   IF OC[R1] = 0                                   CCG27        41     A
SQZ                EQ     "OC",B4,JPX5      IF OC[R1] = OC.LAB                        CCG27        42     A
SQZ                NE     "OC",B5,JPX4      IF OC[R1] " OC.JPX                        CCG27        43     A
SQZ                NZ     X2,JPX4     IF CC$OPF " 0 */ ^OPT=2 & BRIDGE PHASE          CCG27        44     A
SQZ                LX5    -R1.INP                                                     CCG27        45     A
SQZ                SX6    X5          LCI = IN[R1]                                    CCG27        46     A
SQZ                ZR     X6,JPX4     IF LCI = 0                                      CCG27        47     A
SQZ                SA3    B6+X6       LCIA = CFT(LCI)                                 CCG27        48     A
SQZ                SA7    A3          [LCIA] = 0    */ CLEAR FLOW TABLE ENTRY         CCG27        49     A
SQZ       JPX4     SA7    A5+B2       [R1+2] = 0                                      CCG27        50     A
SQZ                EQ     JPX3                                                        CCG27        51     A
SQZ                                                                                   CCG27        52     A
SQZ       JPX5     SA4    A5-B2       DI = R1 - 2                                     CCG27        53     A
SQZ                ZR     X6,MSL1     IF LCI = 0                                      CCG27        54     A
SQZ                NZ     X2,MSL1     IF CC$OPF " 0 */ ^OPT=2 ^ BRIDGE PHASE          CCGA013       5     A
SQZ                SX6    X6+B1                                                       CCG27        55     A
SQZ                SA2    B6+X6       LCIA = CFT(LCI+1)                               CCG27        56     A
SQZ                MI     X2,MSL1     IF JP[LCIA]                                     CCG27        57     A
SQZ                BX6    X2-X3                                                       CCG27        58     A
SQZ                LX6    -30                                                         CCG27        59     A
SQZ                SB3    X6                                                          CCG27        60     A
SQZ                NZ     B3,MSL1     IF FROM"S NE                                    CCG27        61     A
SQZ                SA7    A2          [LCIA+1] = 0  */ CLEAR FALL THROUGH EDGE        CCG27        62     A
SQZ                EQ     MSL1                                                        CCG27        63     A
SQZ                                                                                   CCG27        64     A
SQZ       JPX6     SA2    MRC                                                         CCG27        65     A
SQZ                SX7    A5          JPXA = R1                                       CCG27        66     A
SQZ                LX5    R1.RJP                                                      CCG27        67     A
SQZ                SA7    JPXA                                                        SQZ        2200     A
SQZ                MX4    1                                                           SQZ        2201     A
SQZ                ZR     X2,AUC.RJ   IF MRC = 0                                      SQZ        2202     A
SQZ                BX6    X4+X2                                                       SQZ        2203     A
SQZ                SA6    A4+B1       [DI+1] = 1S59 ! MRC                             SQZ        2204     A
SQZ                SX7    A4+B1                                                       SQZ        2205     A
SQZ                SA7    A2          MRC = DI + 1                                    SQZ        2206     A
SQZ                EQ     AUC.RJ                                                      SQZ        2207     A
SQZ       DEF      SPACE  3                                                           SQZ        2208     A
SQZ                PROCESS DEF                                                        SQZ        2209     A
SQZ                SX3    B1                                                          SQZ        2210     A
SQZ                LX3    D.PSP                                                       SQZ        2211     A
SQZ                BX6    -X3*X4      PS[DI] = 0    */ PS BIT SET IN SBB8 LOOP        SQZ        2212     A
SQZ                SA6    A4                                                          SQZ        2213     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                EQ     MSL1                                                    SQZ        2214     A
SQZ       RJXJ     SPACE  2,14                                                    SQZ        2215     A
SQZ       .PLI     IFNE   .PLI,0                                     ?PL/I        SQZ        2216          I
          -CCG3
SQZ                                                                               SQZ        2217          I
          -CCG3
SQZ                PROCESS RJXJ                                                   SQZ        2218          I
          -CCG3
SQZ                SA3    A5+B1       R2 = R1 + 1                                 SQZ        2219          I
          -CCG3
SQZ                SA1    "RN"+X5     RIW = RND(RI[R1])                           SQZ        2220          I
          -CCG3
SQZ                BX5    X0*X5                                                   SQZ        2221          I
          -CCG3
SQZ                LX3    -IH.RFP                                                 SQZ        2222          I
          -CCG3
SQZ                SA2    "RN"+X3     RFW = RND(RF[R2])                           SQZ        2223          I
          -CCG3
SQZ                BX3    X0*X3                                                   SQZ        2224          I
          -CCG3
SQZ                SX6    X1                                                      SQZ        2225          I
          -CCG3
SQZ                SX7    X2                                                      SQZ        2226          I
          -CCG3
SQZ                IX6    X5+X6       RI[R1] = RN[RIW]                            SQZ        2227          I
          -CCG3
SQZ                SA6    A5                                                      SQZ        2228          I
          -CCG3
SQZ                BX7    X3+X7       RF[R2] = RN[RFW]                            SQZ        2229          I
          -CCG3
SQZ                LX7    IH.RFP                                                  SQZ        2230          I
          -CCG3
SQZ                SA7    A3                                                      SQZ        2231          I
          -CCG3
SQZ                                                                               SQZ        2232          I
          -CCG3
SQZ                SX3    B1                                                      SQZ        2233          I
          -CCG3
SQZ                LX3    RN.USESP                                                SQZ        2234          I
          -CCG3
SQZ                IX6    X1+X3       USES[RIW] = USES[RIW] + 1                   SQZ        2235          I
          -CCG3
SQZ                SA6    A1                                                      SQZ        2236          I
          -CCG3
SQZ                SA2    A2                                                      SQZ        2237          I
          -CCG3
SQZ                IX7    X2+X3       USES[RFW] = USES[RFW] + 1                   SQZ        2238          I
          -CCG3
SQZ                SA7    A2                                                      SQZ        2239          I
          -CCG3
SQZ                EQ     MSL1                                                    SQZ        2240          I
          -CCG3

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                                                                                  SQZ        2241        I
          -CCG3
SQZ        .PLI     ENDIF                                                 'PL/I     SQZ        2242        I
          -CCG3
SQZ       **        RJXJ  RTEST,RFREE,CA,IH  , CA = 12/BIAS,6/JUMP TYPE              CCG3          2    A
SQZ                                                                                  CCG3          3    A
SQZ                 PROCESS RJXJ                                                     CCG3          4    A
SQZ                 SA3    A5+B1       R2 = R1 + 1                                   CCG3          5    A
SQZ                 SA1    "RN"+X5     RIW = RND(RI[R1])                             CCG3          6    A
SQZ                 LX3    -IH.RFP                                                   CCG3          7    A
SQZ                 BX5    X0*X5                                                     CCG3          8    A
SQZ                 SA2    "RN"+X3     RFW = RND(RF[R2])                             CCG3          9    A
SQZ                 BX3    X0*X3                                                     CCG3         10    A
SQZ                 SX6    X1                                                        CCG3         11    A
SQZ                 SX7    X2                                                        CCG3         12    A
SQZ                 IX6    X5+X6       RI[R1] = RN[RIW]                              CCG3         13    A
SQZ                 BX5    X6                                                        CCG3         14    A
SQZ                 SA6    A5                                                        CCG3         15    A
SQZ                 BX7    X3+X7                                                     CCG3         16    A
SQZ                 LX7    IH.RFP      RF[R2] = RN[RFW]                              CCG3         17    A
SQZ                 SA7    A3                                                        CCG3         18    A
SQZ                 LX1    59-RN.CP                                                  CCG3         19    A
SQZ                 PL     X1,RJXJ1    IF ^C[RIW]                                    CCG3         20    A
SQZ                                                                                  CCG3         21    A
SQZ                 SB3    OC.RJ6                                                    CCG3         22        I
          -CCG0002
SQZ        .HC      IFNE   HC.RJ6,0                                                  CCG0002      12    A
SQZ                 SB3    OC.RJ6                                                    CCG0002      13    A
SQZ        .HC      ELSE                                                             CCG0002      14    A
SQZ                 SB3    OC.RJ3                                                    CCG0002      15    A
SQZ        .HC      ENDIF                                                            CCG0002      16    A
SQZ                                                                                  CCG0002      17    A
SQZ                 RJ     ECJ         EVAL CONDITIONAL JUMP                         CCG3         23    A
SQZ                 MI     B5,MSL1     IF TV < 0     */ JUMP CONDITION FALSE         CCG3         24    A
SQZ                 SA5    A5-4        R1 = R1 - 4   */ BACKUP                       CCG3         25    A
SQZ                 EQ     MSL1                                                      CCG3         26    A
SQZ                                                                                  CCG3         27    A
SQZ       *         SEARCH FOR A PREVIOUS OCCURANCE OF THIS INSTRUCTION              CCG3         28    A
SQZ                                                                                  CCG3         29    A
SQZ        RJXJ1    SA3    A5+B1       R2W = [R1+1]                                  CCG3         30    A
SQZ                 SA2    CRJC                                                      CCG3         31    A
SQZ                 SA1    "RN"+X5                                                   CCG3         32    A
SQZ                 BX6    X3                                                        CCG3         33    A
SQZ                 LX1    -RN.TPP                                                   CCG3         34    A
SQZ                 SB5    X1+3        DEFA = TP[RIW] + 3                            CCG3         35    A
SQZ                                                                                  CCG3         36    A
SQZ        RJXJ2    SB4    X2          I = CRJC                                      CCG3         37    A
SQZ                 SA2    X2          CRJC = [I]    */ NEXT                         CCG3         38    A
SQZ                 LE     B4,B5,RJXJ3 IF I @ DEFA                                   CCG3         39    A
SQZ                 SA3    A2-B2       R2C = [I-2]                                   CCG3         40    A
SQZ                 BX7    X6-X3                                                     CCG3         41    A
SQZ                 NZ     X7,RJXJ2    IF R2C " R2W                                  CCG3         42    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                SA3    A3-B1       R1C = [I-3]                                    CCG3    43    A
SQZ                BX7    X3-X5                                                      CCG3    44    A
SQZ                NZ     X7,RJXJ2    IF R1C " R1W                                   CCG3    45    A
SQZ                                                                                  CCG3    46    A
SQZ      *         MATCH FOUND, ELIMINATE THE SECOND INSTRUCTION                     CCG3    47    A
SQZ                                                                                  CCG3    48    A
SQZ                MX7    60                                                         CCG3    49    A
SQZ                SA7    A4          [DI] = 0                                       CCG3    50    A
SQZ                EQ     MSL1                                                       CCG3    51    A
SQZ                                                                                  CCG3    52    A
SQZ      *         INSTRUCTION NOT REDUNDANT, ADD IT TO THE CHAIN, UPDATE *USES*     CCG3    53    A
SQZ                                                                                  CCG3    54    A
SQZ       RJXJ3    SA2    CRJC                                                       CCG3    55    A
SQZ                BX7    X2                                                         CCG3    56    A
SQZ                SA7    A4+B1       [DI+1] = CRJC */ POINT THIS TO LAST            CCG3    57    A
SQZ                SX7    A4+B1                                                      CCG3    58    A
SQZ                SA7    A2          CRJC = DI+1   */ LIST POINTS TO LAST           CCG3    59    A
SQZ                                                                                  CCG3    60    A
SQZ                SA1    "RN"+X5     RIW = RND(RI[R1])                              CCG3    61    A
SQZ                SX4    B1                                                         CCG3    62    A
SQZ                LX4    RN.USESP                                                   CCG3    63    A
SQZ                IX7    X4+X1       USES[RIW] = USES[RIW] + 1                      CCG3    64    A
SQZ                SA7    A1                                                         CCG3    65    A
SQZ                LX6    -IH.RFP                                                    CCG3    66    A
SQZ                SA2    "RN"+X6     RFW = RND(RF[R2])                              CCG3    67    A
SQZ                IX6    X4+X2       USES[RFW] = USES[RFW] + 1                      CCG3    68    A
SQZ                SA6    A2                                                         CCG3    69    A
SQZ                EQ     MSL1                                                       CCG3    70    A
SQZ       ECJ      SPACE  3,14                                                       CCG3    71    A
SQZ      **        ECJ - EVALUATE CONDITIONAL JUMP                                   CCG3    72    A
SQZ      *                                                                           CCG3    73    A
SQZ      *         ENTRY  (X1) = SHIFT(RIW,59-RN.CP)                                 CCG3    74    A
SQZ      *                (B3) = OPC , OPCODE TO BE GENERATED IF JUMP CONDITON      CCG3    75    A
SQZ      *                     IS TRUE                                              CCG3    76    A
SQZ      *                                                                           CCG3    77    A
SQZ      *         EXIT   (B5) = TV, TRUTH VALUE, < 0 IF JUMP CONDITION IS FALSE     CCG3    78    A
SQZ      *                (X1) = R1S , OLD VALUE OF R1 WORD                          CCG3    79    A
SQZ                                                                                  CCG3    80    A
SQZ       ECJ3     MX7    60                                                         CCG3    81    A
SQZ                SA1    A5                                                         CCG3    82    A
SQZ                SB5    -B1         TV = -1       */ INDICATE CONDTION FALSE       CCG3    83    A
SQZ                SA7    A4          [DI] = -0     */ MARK AS REDUNDANT             CCG3    84    A
SQZ                                                                                  CCG3    85    A
SQZ       ECJ      ROUTINE                                                           CCG3    86    A
SQZ                LX1    1+RN.CP-RN.TPP                                             CCG3    87    A
SQZ                SA3    X1                                                         CCG3    88    A
SQZ                RJ     DCV         CV = DCV( TP[RIW] )                            CCG3    89    A
SQZ                SA3    A5+B1       R2W = [R1+1]                                   CCG3    90    A
SQZ                MX1    -3                                                         CCG3    91    A
SQZ                LX3    -IH.CAP                                                    CCG3    92    A
SQZ                BX2    -X1*X3                                                     CCG3    93    A
SQZ                SB5    X2          TV = JT[CA[R2W]]                               CCG3    94    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                JP     ECJ1+B5                                                CCG3          95     A
SQZ                                                                              CCG3          96     A
SQZ        ECJ1    BSS    0                                                      CCG3          97     A
SQZ                ECHO   2,XX=("XJUMPS")                                        CCG3          98     A
SQZ                XX     X7,ECJ2                                                CCG3          99     A
SQZ                EQ     ECJ3                                                   CCG3         100     A
SQZ                                                                              CCG3         101     A
SQZ       *        JUMP CONDITION IS TRUE, CHANGE INST TO A *UJP/RJ6*             CCG3         102     A
SQZ                                                                              CCG3         103     A
SQZ        ECJ2    LX3    -6                                                     CCG3         104     A
SQZ                MX1    -12                                                    CCG3         105     A
SQZ                BX6    -X1*X3                                                 CCG3         106     A
SQZ                LX6    R1.CAP                                                 CCG3         107     A
SQZ                LX3    IH.CAP+6-IH.IHP                                        CCG3         108     A
SQZ                MX1    -IH.IHL                                                CCG3         109     A
SQZ                BX2    -X1*X3                                                 CCG3         110     A
SQZ                BX6    X6+X2                                                  CCG3         111     A
SQZ                SA1    A5          R1S = [R1]                                 CCG3         112     A
SQZ                SB6    B0                                                     CCG27         68     A
SQZ                RJ     URI         CHANGE INSTRUCTION TO A UJP/RJ6            CCG3         113     A
SQZ                MX7    0                                                      CCG3         114     A
SQZ                SA7    A5+B1                                                  CCG3         115     A
SQZ                EQ     ECJ                                                    CCG3         116     A
SQZ        SMR     TITLE  SMR - SQUEEZE MEMORY REFERENCES                        SQZ         2243     A
SQZ       **       MEMORY REFERENCE PROCESSING                                   SQZ         2244     A
SQZ                                                                              SQZ         2245     A
SQZ                PROCESS (TLD,TST,ILD)                                         SQZ         2246          I
          -CCG16
SQZ                PROCESS (TLD,TST,ILD,LDV)                                     CCG16        251     A
SQZ                SA3    A5+B1       R2 = R1 + 1                                SQZ         2247     A
SQZ                SB5    "TXT"+3     DEFA = TXT + 3  */  MIN SEARCH ADDRESS     SQZ         2248     A
SQZ                BX6    X3          R2W = [R2]                                 SQZ         2249     A
SQZ                EQ     SMR                                                    SQZ         2250     A
SQZ        DRL     SPACE  2                                                      FCC2072        7     A
SQZ        .LCM    IFNE   .DAL,0                                                 FCC2072        8     A
SQZ                PROCESS (DRL,DWL)                                             FCC2072        9     A
SQZ                LX1    -RN.TPP                                                FCC2072       10     A
SQZ                SA2    X1          R1J = TP[RJW]                              FCC2072       11     A
SQZ                SB5    X1+3        DEFA = TP[RJW] + 3                         FCC2072       12     A
SQZ                SA3    A5+B1       R2 = R1 + 1                                FCC2072       13     A
SQZ                UX6    B4,X2                                                  FCC2072       14     A
SQZ                SX6    B4-OC.STT                                              FCC2072       15     A
SQZ                NZ     X6,SLD1     IF OC[R1J] " OC.STT                        FCC2072       16     A
SQZ                SA3    A2+B1       R2 = R14 + 1  */ SUBSTITUTE R2 OF STT      FCC2072       17     A
SQZ                EQ     SLD1                                                   FCC2072       18     A
SQZ        .LCM    ENDIF                                                         FCC2072       19     A
SQZ        PLD     SPACE  2                                                      SQZ         2251     A
SQZ                PROCESS (PLD,PST)                                             SQZ         2252     A
SQZ                LX1    -RN.TPP                                                SQZ         2253     A
SQZ                SA3    X1+B2       DJ = RP[RJW] + 2                           SQZ         2254     A
SQZ                SX6    1                                                      SQZ         2255     A
SQZ                LX6    D.RFP                                                  SQZ         2256     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                BX7    X6+X3       RF[DJ] = 1                                  SQZ        2257    A
SQZ                LX1    RN.TPP                                                  SQZ        2258    A
SQZ                SA7    A3                                                      SQZ        2259    A
SQZ        SLD     SPACE  2                                                       SQZ        2260    A
SQZ                PROCESS (SLD,SST,SDL,SDS,DRL,DWL)                              SQZ        2261         I
           -FCC2072
SQZ                PROCESS (SLD,SST,SDL,SDS)                                      FCC2072      20    A
SQZ                SA3    A5+B1       R2 = R1 + 1                                 SQZ        2262    A
SQZ                LX1    -RN.TPP                                                 SQZ        2263    A
SQZ                SB5    X1+3        DEFA = TP[RJW] + 3                          SQZ        2264    A
SQZ                LX3    -IH.RFP                                                 SQZ        2265         I
           -FCC2072
SQZ        SLD1    LX3    -IH.RFP                                                 FCC2072      21    A
SQZ                SA2    "RN"+X3     RFW = RND(RF[R2])                           SQZ        2266         I
           -FCC0643
SQZ                BX7    -X0*X3                                                  FCC0643      79    A
SQZ                SA2    "RN"+X7     RFW = RND(RF[R2])                           FCC0643      80    A
SQZ                BX6    X0*X3                                                   SQZ        2267    A
SQZ                S"OC"  B0          OC = 0        */ INDICATE SHORT MEM REF    SQZ        2268    A
SQZ                SX7    X2                                                      SQZ        2269    A
SQZ                BX6    X6+X7       RF[R2] = RN[RFW]                            SQZ        2270    A
SQZ                LX6    IH.RFP      R2W = [R2]                                  SQZ        2271    A
SQZ                SA6    A5+B1                                                   SQZ        2272    A
SQZ                EQ     SMR                                                     SQZ        2273    A
SQZ        LD      SPACE  2                                                       SQZ        2274    A
SQZ                PROCESS (LD,ST)                                                SQZ        2275    A
SQZ                SA2    A5+B1       R2 = R1 + 1                                 SQZ        2276    A
SQZ                LX2    -IH.RFP                                                 SQZ        2277         I
           -FCC0792
SQZ                SA1    "RN"+X2     RFW = RND(RF[R2])                           SQZ        2278         I
           -FCC0792
SQZ                MX7    -RN.CIAL                                                SQZ        2279         I
           -FCC0792
SQZ                BX6    X0*X2       R2W = [R2]                                  SQZ        2280         I
           -FCC0792
SQZ                LX7    RN.CIAP                                                 SQZ        2281         I
           -FCC0792
SQZ                BX3    -X7*X1                                                  SQZ        2282         I
           -FCC0792
SQZ                SX7    X1          RF[R2W] = RN[RFW]                           SQZ        2283         I
           -FCC0792
SQZ                ZR     X3,LD1      IF ^(C,SA,IAC)[RFW]                         SQZ        2284         I
           -FCC0792
SQZ                SA3    ARFA                                                    FCC0792      11    A
SQZ                LX2    -IH.RFP                                                 FCC0792      12    A
SQZ                SA1    "RN"+X2     RFW = RND(RF[R2])                           FCC0792      13    A
SQZ                BX6    X0*X2       R2W = [R2]                                  FCC0792      14    A
SQZ                SX7    X1          RF[R2W] = RN[RFW]                           FCC0792      15    A
SQZ                BX3    X3*X1                                                   FCC0792      16    A
SQZ                ZR     X3,LD1      IF ^(C,SA,IAC,XMT)[RFW]                     FCC0792      17    A
SQZ                RJ     ARF         ADJUST RF,CA                                SQZ        2285    A
SQZ                                                                               SQZ        2286    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ       LD1      BX6    X6+X7                                                  SQZ        2287    A
SQZ                LX6    IH.RFP                                                 SQZ        2288    A
SQZ                SA6    A5+1        [R2] = R2W                                 SQZ        2289         I
          -FCC1029
SQZ                SA6    A5+B1       [R2] = R2W                                 FCC1029       5    A
SQZ                LX1    -RN.TPP                                                SQZ        2290    A
SQZ                SB5    X1+3        DEFA = TP[RFW] + 3  */ MIN SRCH ADDRESS    SQZ        2291    A
SQZ       *        EQ     SMR                                                    SQZ        2292         I
          -FCC1029
SQZ                ERRNZ  59-IH.LDP                                              FCC1029       6    A
SQZ                PL     X6,SMR      IF ^LD[R2W]   */ ^PREFETCH ! APL *LD*      FCC1029       7    A
SQZ                                                                              FCC1029       8    A
SQZ                LX4    59-D.STP                                               FCC1029       9    A
SQZ                EQ     SMR2                                                   FCC1029      10    A
SQZ       SMR      EJECT                                                         SQZ        2293    A
SQZ       **       SMR - SQUEEZE MEMORY REFERENCES                               SQZ        2294    A
SQZ       *                                                                      SQZ        2295    A
SQZ       *        ENTRY  (X6) = R2W , R2 WORD OF INSTRUCTION WITH *RF* ADJUSTED SQZ        2296    A
SQZ       *               (B5) = DEFA, MIN SEARCH ADDRESS                        SQZ        2297    A
SQZ       *               (B3) = 0 IF A TYPE I MEM REF, ELSE *OC*                SQZ        2298    A
SQZ       *               (A4,X4,A5,X5) AS FROM MAIN LOOP                        SQZ        2299    A
SQZ                                                                              SQZ        2300    A
SQZ       SMR      SA2    MRC                                                    SQZ        2301    A
SQZ                LX4    59-D.STP                                               SQZ        2302    A
SQZ                MX1    -IH.IHL                                                SQZ        2303    A
SQZ                                                                              SQZ        2304    A
SQZ       *        SEARCH *MRC* BACKWARDS FOR REF TO SAME MEMORY LOCATION         SQZ        2305    A
SQZ       *        OR INTERFERING STORE INTO THE SAME *IH* CLASS                  SQZ        2306    A
SQZ       *        NOTE THAT ONLY THE *IH* WORDS ARE CHECKED.                     SQZ        2307    A
SQZ                                                                              SQZ        2308    A
SQZ       SMR1     SB4    X2          I = MRC                                    SQZ        2309    A
SQZ                SA2    X2          MRC = [MRC]                                SQZ        2310    A
SQZ                LE     B4,B5,SMR2  IF I @ DEFA   */ END OF CHAIN              SQZ        2311    A
SQZ                BX7    X2*X4                                                  SQZ        2312    A
SQZ                SA3    A2-B2       R2M = MRC - 2                              SQZ        2313    A
SQZ                MI     X7,SMR2     IF JP[MRC] & ST[DI]                        SQZ        2314    A
SQZ                BX7    X3-X6                                                  SQZ        2315    A
SQZ                ZR     X7,SMR7     IF [R2M] = R2W  */ REFS TO SAME MEMORY LOC SQZ        2316    A
SQZ                                                                              SQZ        2317    A
SQZ       *        CHECK FOR DATA INTERFERENCE                                    SQZ        2318    A
SQZ                                                                              SQZ        2319    A
SQZ                LX7    -IH.IHP                                                SQZ        2320    A
SQZ                BX3    -X1*X7                                                 SQZ        2321    A
SQZ                NZ     X3,SMR1     IF IH[R2M] " IH[R2W]                       SQZ        2322    A
SQZ                LX7    IH.IHP-IH.RFP                                          SQZ        2323    A
SQZ                BX7    -X0*X7                                                 SQZ        2324    A
SQZ                SA3    A3+B1                                                  SQZ        2325    A
SQZ                ZR     X7,SMR1     IF RF[R2M] = RF[R2W]                       SQZ        2326    A
SQZ                LX3    59-D.STP                                               SQZ        2327    A
SQZ                BX7    X3+X4                                                  SQZ        2328    A
SQZ                PL     X7,SMR1     IF ^( ST[DI] ! ST[DM] )                    SQZ        2329    A
SQZ       SMR      SPACE  2                                                      SQZ        2330    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ       *         LD/ST NOT REDUNDANT, ADD TO *MRC*, AND BUMP USES OF OPERANDS          SQZ        2331    A
SQZ                                                                                       SQZ        2332    A
SQZ        SMR2     SA2    MRC                                                            SQZ        2333    A
SQZ                 SA3    A4+B1                                                          SQZ        2334    A
SQZ                 LX6    -IH.RFP                                                        SQZ        2335    A
SQZ                 BX7    X2+X3       LINK[DI+1] = MRC                                   SQZ        2336    A
SQZ                 SA7    A3                                                             SQZ        2337    A
SQZ                 SB5    X6          R = RF[R2W]                                        SQZ        2338    A
SQZ                 SX7    A4+B1                                                          SQZ        2339    A
SQZ                 SX6    1                                                              SQZ        2340    A
SQZ                 SA7    A2          MRC = DI + 1  */ CHAIN POINTS TO LAST              SQZ        2341    A
SQZ                 LX6    RN.USESP                                                       SQZ        2342    A
SQZ                 ZR     B3,SMR5     IF OC = 0     */ TYPE I MEM REF                    SQZ        2343    A
SQZ        SMR      SPACE  2,12                                                           SQZ        2344    A
SQZ       *         ADVANCE USES FOR TYPE III MEM REF                                     SQZ        2345    A
SQZ                                                                                       SQZ        2346    A
SQZ                 SA1    "RN"+B5     RFW = RND(R)                                       SQZ        2347    A
SQZ                 IX7    X6+X1       USES[RFW] = USES[RFW] + 1                          SQZ        2348    A
SQZ                 LX1    -RN.TPP                                                        SQZ        2349    A
SQZ                 SA7    A1                                                             SQZ        2350    A
SQZ                 LE     B5,B1,SMR3        IF R @ 1                                     SQZ        2351    A
SQZ                                                                                       SQZ        2352    A
SQZ        .LCM     IFNE   .DAL,0                                                         CCG21        69    A
SQZ                 SA1    LCM#G                                                          CCG21        70          I
          -CCG0012
SQZ                 BX7    X1*X4                                                          CCG21        71          I
          -CCG0012
SQZ                 SA2    LCM#G                                                          CCG0012      13    A
SQZ                 BX7    X2*X4                                                          CCG0012      14    A
SQZ                 NZ     X7,SMR3     IF LCM = G AND L2[DI]  */ 21 BIT SUBSCRIPT         CCG21        72    A
SQZ        .LCM     ENDIF                                                                 CCG21        73    A
SQZ                 SA2    X1+B2       DIP = TP[RFW] + 2                                  SQZ        2353    A
SQZ                 SX3    B1                                                             SQZ        2354    A
SQZ                 LX3    D.RFP                                                          SQZ        2355    A
SQZ                 BX7    X3+X2       RF[DIP] = 1                                        SQZ        2356    A
SQZ                 SA7    A2                                                             SQZ        2357    A
SQZ                                                                                       SQZ        2358    A
SQZ        SMR3     MI     X4,SMR4     IF ST[DI]                                          SQZ        2359    A
SQZ                                                                                       SQZ        2360    A
SQZ                 GT     B5,B1,MSL1  IF R > 1                                           SQZ        2361    A
SQZ                 LX4    1+D.STP                                                        SQZ        2362    A
SQZ                 LX6    D.ZPP-RN.USESP                                                 SQZ        2363    A
SQZ                 BX7    X6+X4       ZP[DI] = 1                                         SQZ        2364    A
SQZ                 SA7    A4                                                             SQZ        2365    A
SQZ                 EQ     MSL1                                                           SQZ        2366    A
SQZ                                                                                       SQZ        2367    A
SQZ       *         ADVANCE USES FOR *RI* IN STORE INSTRUCTION                            SQZ        2368    A
SQZ                                                                                       SQZ        2369    A
SQZ        SMR4     SA1    "RN"+B6                                                        SQZ        2370    A
SQZ                 BX3    X0*X5                                                          SQZ        2371    A
SQZ                 SA1    "RN"+X1     RIW = RND(RI)                                      SQZ        2372    A
SQZ                 SX2    X1                                                             SQZ        2373    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                IX7    X6+X1       USES[RIW] = USES[RIW] + 1                    SQZ        2374    A
SQZ                SA7    A1                                                       SQZ        2375    A
SQZ                BX6    X3+X2       RI[R1] = RN[RIW]                             SQZ        2376    A
SQZ                SA6    A5                                                       SQZ        2377    A
SQZ                LX4    D.STP-D.L2P                                              SQZ        2378         I
          -FCC0957
SQZ                MI     X4,MSL1     IF L2[DI]                                    SQZ        2379         I
          -FCC0957
SQZ                SX7    A4+B1                                                    SQZ        2380         I
          -FCC0957
SQZ                SA7    LSTA        LSTA = DI + 1                                SQZ        2381         I
          -FCC0957
SQZ                EQ     MSL1                                                     SQZ        2382    A
SQZ                                                                               SQZ        2383    A
SQZ      *         ADVANCE USES FOR TYPE I MEM REF                                SQZ        2384    A
SQZ                                                                               SQZ        2385    A
SQZ       SMR5     LX5    -R1.RJP                                                  SQZ        2386    A
SQZ                BX7    -X0*X5                                                   SQZ        2387    A
SQZ                SA1    "RN"+X7     RNJ = RND[RJ[R1])                            SQZ        2388    A
SQZ                LX5    R1.RJP-R1.RKP                                            SQZ        2389    A
SQZ                IX7    X6+X1       USES[RNJ] = USES[RNJ] + 1                    SQZ        2390    A
SQZ                SA7    A1                                                       SQZ        2391    A
SQZ                BX3    -X0*X5                                                   SQZ        2392    A
SQZ                SA1    "RN"+X3     RNK = RND(RK[R1])                            SQZ        2393    A
SQZ                IX7    X6+X1       USES[RNK] = USES[RNK] + 1                    SQZ        2394    A
SQZ                SA7    A1                                                       SQZ        2395    A
SQZ                PL     X4,MSL1     IF ^ST[DI]                                   SQZ        2396    A
SQZ                                                                               SQZ        2397    A
SQZ                LX5    R1.RKP                                                   SQZ        2398    A
SQZ                EQ     SMR4                                                     SQZ        2399    A
SQZ       SMR      SPACE  2,10                                                     SQZ        2400    A
SQZ      *         2 MEMORY REFERENCES TO THE SAME LOCATION                       SQZ        2401    A
SQZ                                                                               SQZ        2402    A
SQZ       SMR7     SA1    A3-B1       R1M = R2M - 1                                SQZ        2403    A
SQZ                SA3    A3+B1       DM = R2M + 1                                 SQZ        2404    A
SQZ                SB4    OC.ILD                                                   SQZ        2405    A
SQZ                LX3    59-D.STP                                                 SQZ        2406    A
SQZ                MI     X4,SMR11    IF ST[DI]                                    SQZ        2407    A
SQZ                MI     X3,SMR9     IF ST[DM]                                    SQZ        2408    A
SQZ                                                                               SQZ        2409    A
SQZ      *         LD / LD - ELIMINATE THE SECOND                                 SQZ        2410    A
SQZ                                                                               SQZ        2411    A
SQZ                NE     "OC",B4,SMR8      IF OC[R1] " OC.ILD                     SQZ        2412    A
SQZ                SA2    JPXA                                                     SQZ        2413    A
SQZ                SX6    A3                                                       SQZ        2414    A
SQZ                IX7    X6-X2                                                    SQZ        2415    A
SQZ                MI     X7,SMR8     IF DM < JPXA  */ FIRST IS BEFORE LAST *JPX*  SQZ        2416    A
SQZ                UX7    "OC",X1                                                  SQZ        2417    A
SQZ                SX6    "OC"-OC.LD                                               SQZ        2418    A
SQZ                NZ     X6,SMR8     IF OC[R1M] " OC.LD                           SQZ        2419    A
SQZ                                                                               SQZ        2420    A
SQZ                PX7    B4,X1       OC[R1M] = OC.ILD                             SQZ        2421    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                SA7    A1                                                    SQZ        2422    A
SQZ                                                                             SQZ        2423    A
SQZ       SMR8     BX7    X3*X4                                                 SQZ        2424    A
SQZ                LX7    D.STP-D.PRSP                                          SQZ        2425    A
SQZ                MI     X7,SMR10    IF PRS[DM] & PRS[DI]                      SQZ        2426    A
SQZ                                                                             SQZ        2427    A
SQZ      *         KILL THE SECOND AND OR IN THE PRECEDENCE BITS                SQZ        2428    A
SQZ                                                                             SQZ        2429    A
SQZ                BX7    X3+X4                                                 SQZ        2430    A
SQZ       #DAL     IFNE   .DAL,0                                                SQZ        2431    A
SQZ                MX3    -D.STRSL                                              SQZ        2432    A
SQZ                LX3    D.STRSP+59-D.STP                                      SQZ        2433    A
SQZ                BX6    -X3*X7                                                SQZ        2434    A
SQZ                NZ     X6,SMR10    IF STRS[DI] ! STRS[DM]                    SQZ        2435    A
SQZ       #DAL     ENDIF                                                        SQZ        2436    A
SQZ                MX6    0                                                     SQZ        2437    A
SQZ                LX7    1+D.STP                                               SQZ        2438    A
SQZ                SA7    A3          DM = DM ! DI                              SQZ        2439    A
SQZ                SA6    A4          [DI] = 0                                  SQZ        2440    A
SQZ                SA1    "RN"+X1                                               SQZ        2441    A
SQZ                BX7    X1                                                    SQZ        2442    A
SQZ                SA7    "RN"+B6     RND(RI) = RND(RI[R1M])                    SQZ        2443    A
SQZ                EQ     MSL1                                                  SQZ        2444    A
SQZ       SMR      SPACE  2,12                                                  SQZ        2445    A
SQZ      *         ST / LD - CHANGE TO AN *XMT*                                 SQZ        2446    A
SQZ                                                                             SQZ        2447    A
SQZ       SMR9     SA6    STLDS       STLDS = R2W   */ SET FLAG FOR *AUV*       SQZ        2448    A
SQZ                NE     B3,B4,SMR10 IF OC[R1] " OC.ILD                        SQZ        2449    A
SQZ                SA2    JPXA                                                  SQZ        2450    A
SQZ                SX6    A3                                                    SQZ        2451    A
SQZ                IX7    X6-X2                                                 SQZ        2452    A
SQZ                MI     X7,SMR10    IF DM < JPXA  */ *ST* BEFORE LAST *JPX*   SQZ        2453    A
SQZ                                                                             SQZ        2454    A
SQZ                RJ     EFS         ELIMINATE FIRST STORE                     SQZ        2455    A
SQZ                                                                             SQZ        2456    A
SQZ       SMR10    BX2    -X0*X1      RJ = RI[R1M]                              SQZ        2457    A
SQZ                LX4    1+D.STP                                               SQZ        2458    A
SQZ                SA1    "RN"+X2     (X1) = RND(RJ) FOR CTI                    SQZ        2459    A
SQZ                MX7    0                                                     SQZ        2460    A
SQZ                SA7    A5+B1       [R2] = 0                                  SQZ        2461    A
SQZ                EQ     SRCH6       CHANGE THE *LD* TO AN *XMT*               SQZ        2462         I
          -FCC0509
SQZ                                                                             FCC0509      47    A
SQZ      *         SET BIT IN *SUVA* FOR REMOVED *LD* IF CALLED FROM GPO/AUV.   FCC0509      48    A
SQZ                                                                             FCC0509      49    A
SQZ                SA2    SUVA                                                  FCC0509      50    A
SQZ                ZR     X2,SMR10A   IF SUVA = 0   */ NOT CALL FROM *AUV*      FCC0509      51    A
SQZ                LX5    -R1.INP                                               FCC0509      52    A
SQZ                SB5    X5+B1       I = IN[R1]                                FCC0509      53    A
SQZ                EQ     B5,B1,SMR10A      IF I = 0  */ NO *UDT* INDEX         FCC0509      54    A
SQZ                SA3    =XO.UDT                                               FCC0509      55         I
          -CCGNC

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                SA3    =XO$UDT                                                  CCGNC        764     A
SQZ                SB4    X2                                                       FCC0509       56     A
SQZ                SA3    X3+B5       U2 = UDT(I+1)                                FCC0509       57     A
SQZ                UX7    B3,X3       (B,WI) = BIT,WI)[U2]                         FCC0509       58     A
SQZ                SA2    B4+X3                                                    FCC0509       59     A
SQZ                SX6    B1                                                       FCC0509       60     A
SQZ                LX6    B3,X6                                                    FCC0509       61     A
SQZ                BX7    X6+X2       B[SUVA(WI)] = 1                              FCC0509       62     A
SQZ                SA7    A2                                                       FCC0509       63     A
SQZ                                                                               FCC0509       64     A
SQZ       SMR10A   SX2    X1          RJ = RN[RND(RJ)]                             FCC0509       65     A
SQZ                EQ     SRCH6             */ CHANGE *LD* TO AN *XMT*             FCC0509       66     A
SQZ       SMR      SPACE  3,14                                                     SQZ         2463     A
SQZ       SMR11    MI     X3,SMR12    IF ST[DM]                                    SQZ         2464     A
SQZ                                                                               SQZ         2465     A
SQZ      *         LD / ST - CHECK FOR A = A                                       SQZ         2466     A
SQZ                                                                               SQZ         2467     A
SQZ                SA2    "RN"+B6                                                  SQZ         2468     A
SQZ                SA2    "RN"+X2     RI = RND(RI)                                 SQZ         2469     A
SQZ                LX2    59-RN.XMTP                                               SQZ         2470     A
SQZ                PL     X2,SMR2     IF ^XMT[RI]                                  SQZ         2471     A
SQZ                LX2    1+RN.XMTP-RN.TPP                                         SQZ         2472     A
SQZ                SA2    X2          R1X = TP[RND(RI)]                            SQZ         2473     A
SQZ                LX2    -R1.RJP                                                  SQZ         2474     A
SQZ                BX3    X1-X2                                                    SQZ         2475     A
SQZ                BX7    -X0*X3                                                   SQZ         2476     A
SQZ                NZ     X7,SMR2     IF RJ[R1X] " RI[RIM]                         SQZ         2477     A
SQZ                SA7    A4          [DI] = 0   ( KILL SECOND STORE )             SQZ         2478     A
SQZ                EQ     MSL1                                                     SQZ         2479     A
SQZ                                                                               SQZ         2480     A
SQZ      *         ST / ST - ELIMINATE THE FIRST STORE                             SQZ         2481     A
SQZ                                                                               SQZ         2482     A
SQZ       SMR12    RJ     EFS         ELEMINATE FIRST STORE                        SQZ         2483     A
SQZ                EQ     SMR2                                                     SQZ         2484     A
SQZ       EFS      TITLE  EFS - ELIMINATE FIRST STORE, ADJUST *PS* BIT             SQZ         2485          I
         -FCC0957
SQZ       EFS      SPACE  3                                                        FCC0957        5     A
SQZ      **        EFS - ELIMINATE FIRST STORE, ADJUST *PS* BIT FOR *RI* IN ST     SQZ         2486     A
SQZ      *                                                                        SQZ         2487     A
SQZ      *         ENTRY  (X1) = RIM , R1 OF FIRST STORE                           SQZ         2488     A
SQZ      *                (A3) = DM , DESCRIPTOR ADDR                              SQZ         2489     A
SQZ      *                                                                        SQZ         2490     A
SQZ      *         EXIT   (X1) = R , RI[R1M]                                       SQZ         2491     A
SQZ      *                (X6) = 0                                                 SQZ         2492     A
SQZ                                                                               SQZ         2493     A
SQZ       EFS      ROUTINE                                                         SQZ         2494     A
SQZ                BX1    -X0*X1      R = RI[R1M]                                  SQZ         2495     A
SQZ                SA2    "RN"+X1     RNI = RND(R)                                 SQZ         2496     A
SQZ                MX6    60          D = [DM]                                     SQZ         2497     A
SQZ                SX7    B1                                                       SQZ         2498     A
SQZ                SA6    A3          [DM] = -0     */ KILL FIRST *ST*             SQZ         2499     A
SQZ                ZR     X3,EFS      IF D = 0      */   ST,ILD,ILD SITUATION       SQZ         2500     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                LX7    RN.USESP                                                  SQZ        2501     A
SQZ                IX7    X2-X7       USES[RNI] = USES[RNI] - 1                     SQZ        2502     A
SQZ                SA7    A2                                                        SQZ        2503     A
SQZ                BX3    X4                                                        FCC0957       6     A
SQZ                LX3    D.STP-D.L2P                                               FCC0957       7     A
SQZ                MI     X3,EFS      IF L2[DI]     */ PS BIT NOT SET               FCC0957       8     A
SQZ                LX2    -RN.TPP                                                   SQZ        2504     A
SQZ                MX3    -RN.USESL                                                 SQZ        2505          I
          -FCC0957
SQZ                SB4    X2          DEFA = TP[RNI]  */ R1 ADDR OF DEF             SQZ        2506     A
SQZ                LX7    -RN.USESP                                                 SQZ        2507          I
          -FCC0957
SQZ                BX7    -X3*X7                                                    SQZ        2508          I
          -FCC0957
SQZ                ZR     X7,EFS3     IF USES[RNI] = 0                              SQZ        2509          I
          -FCC0957
SQZ                                                                                SQZ        2510          I
          -FCC0957
SQZ       *        SEARCH PRECEEDING STORES TO SEE IF *RI* USED BY ANOTHER *ST*    SQZ        2511          I
          -FCC0957
SQZ                                                                                SQZ        2512          I
          -FCC0957
SQZ                SB5    A4+B1       LINK = DI + 1                                 SQZ        2513          I
          -FCC0957
SQZ                MI     X4,EFS1     IF ST[DI]     */ NOT *ILD*                    SQZ        2514          I
          -FCC0957
SQZ                SA2    LSTA                                                      SQZ        2515          I
          -FCC0957
SQZ                SB5    X2          LINK = LSTA   */ USE ADDR OF LAST ST          SQZ        2516          I
          -FCC0957
SQZ                ZR     X2,EFS3     IF LINK = 0                                   SQZ        2517          I
          -FCC0957
SQZ                SA3    STC                                                       FCC0957       9     A
SQZ                SB5    X3          LINK = STC                                    FCC0957      10     A
SQZ                                                                                FCC0957      11     A
SQZ       *        SEARCH *STC* TO SEE IF *RI* IS USED BY ANOTHER *ST*             FCC0957      12     A
SQZ                                                                                SQZ        2518     A
SQZ        EFS1    SA2    B5-B1       DL = LINK - 1                                 SQZ        2519     A
SQZ                SA3    B5          LW = [LINK]                                   SQZ        2520     A
SQZ                ZR     X2,EFS2     IF [DL] = 0   */ DEAD INSTRUCTION             SQZ        2521     A
SQZ                SA2    A2-2        R1L = DL - 2                                  SQZ        2522          I
          -FCC0957
SQZ                SA2    A2-B2       R1L = DL - 2                                  FCC0957      13     A
SQZ                BX7    -X0*X2                                                    FCC0957      14     A
SQZ                SA2    "RN"+X7     RL = RN[RND(RI[R1L])]                         FCC0957      15     A
SQZ                BX7    X2-X1                                                     SQZ        2523     A
SQZ                BX2    -X0*X7                                                    SQZ        2524     A
SQZ                ZR     X2,EFS      IF RI[R1L] = R  */ USED BY ANOTHER ST         SQZ        2525          I
          -FCC0957
SQZ                ZR     X2,EFS      IF RL = R     */ USED BY ANOTHER *ST*         FCC0957      16     A
SQZ        EFS2    LX3    -I.STCP                                                   SQZ        2526     A
SQZ                SB5    X3          LINK = STC[LW]                                SQZ        2527     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                NZ     B5,EFS1     IF LINK " 0                                    SQZ         2528          I
          -FCC0957
SQZ                                                                                  SQZ         2529          I
          -FCC0957
SQZ       *        NO OTHER PREVIOUS IN A *ST* ON *STC*, CHECK THIS *ST*             SQZ         2530          I
          -FCC0957
SQZ                                                                                  SQZ         2531          I
          -FCC0957
SQZ        EFS3    SA3    "RN"+B6     RNI = RND(RI[R1])                              SQZ         2532          I
          -FCC0957
SQZ                BX7    X3-X1                                                      SQZ         2533          I
          -FCC0957
SQZ                BX2    -X0*X7                                                     SQZ         2534          I
          -FCC0957
SQZ                SX7    B1                                                         SQZ         2535          I
          -FCC0957
SQZ                ZR     X2,EFS      IF RN[RNI] = R  */ USED BY THIS INST           SQZ         2536          I
          -FCC0957
SQZ                GT     B5,B4,EFS1  IF LINK > DEFA                                 FCC0957       17     A
SQZ                                                                                  SQZ         2537     A
SQZ                SA2    B4+B2       DIP = DEFA + 2  */ DESCR OF DEF                SQZ         2538     A
SQZ                SX7    B1                                                         FCC0957       18     A
SQZ                LX7    D.PSP                                                      SQZ         2539     A
SQZ                BX7    -X7*X2      PS[DIP] = 0                                    SQZ         2540     A
SQZ                SA7    A2                                                         SQZ         2541     A
SQZ                EQ     EFS                                                        SQZ         2542     A
SQZ        ARF     TITLE  ARF - ADJUST RF,CA OF INDEXED MEM REFERENCE                SQZ         2543     A
SQZ       **       ARF - ADJUST RF,CA OF INDEXED MEMORY REFERENCE TO FOLD IN         SQZ         2544     A
SQZ       *                CONSTANT PART OF *RF*                                    SQZ         2545     A
SQZ       *                                                                         SQZ         2546     A
SQZ       *        ENTRY  (X1) = RND(RF)                                             SQZ         2547     A
SQZ       *                (X3) = (C,SA,IAC)[RFW]                                   SQZ         2548     A
SQZ       *                (X6) = SHIFT( R2W , -IH.RFP ) AND RF[R2W] = 0            SQZ         2549     A
SQZ       *                (X7) = RF                                                SQZ         2550     A
SQZ       *                                                                         SQZ         2551     A
SQZ       *        EXIT   *RF* ADJUSTED AND X1,X6,X7 AS ON ENTRY                     SQZ         2552     A
SQZ                                                                                  FCC0792       18     A
SQZ        ARFA    BFMW   RN,(C,SA,IAC,XMT)                                          FCC0792       19     A
SQZ        ARFAIL  BSS    1                                                  002601  CCG37          4     A
SQZ                                                                                  SQZ         2553     A
SQZ        ARF1    SA3    X1          R1P = TP[RFW]                                  SQZ         2554     A
SQZ                RJ     DCV         GET CONSTANT VALUE                             SQZ         2555          I
          -CCG20
SQZ                RJ     DCV         SVAL = DCV(R1P)                                CCG20          2     A
SQZ                BX3    X7                                                         CCG20          3          I
          -CCG24
SQZ                                                                                  CCG24         24     A
SQZ        ARF0    BX3    X7                                                         CCG24         25     A
SQZ                AX3    17                                                         CCG20          4     A
SQZ                NZ     X3,ARF7     IF SHIFT(SVAL,-17) " 0  */ FULL WORD CONSTANT  CCG20          5     A
SQZ                                                                                  SQZ         2556     A
SQZ        ARF0    LX6    IH.RFP-IH.CAP                                              SQZ         2557          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

          -CCG24
SQZ                 LX6    IH.RFP-IH.CAP                                    CCG24        26     A
SQZ                 SB4    X7                                               SQZ        2558     A
SQZ       #DAL      IFNE   .DAL,0                                           CCG20         6     A
SQZ                 LX4    17-D.L2P                                         CCG20         7     A
SQZ                 BX7    X4*X6                                            CCG20         8     A
SQZ                 LX4    D.L2P-17                                         CCG20         9     A
SQZ                 MI     X7,ARF7     IF L2[DI] & IND[CA[R2]]              CCG20        10     A
SQZ       #DAL      ENDIF                                                    CCG20        11     A
SQZ                 SX3    X6+B4       CAV = CA[R2] + SVAL                  SQZ        2559     A
SQZ                 MX7    -IH.CAL                                          SQZ        2560     A
SQZ                 BX3    -X7*X3                                           SQZ        2561     A
SQZ                 BX7    X7*X6                                            SQZ        2562     A
SQZ                 BX6    X3+X7       CA[R2] = CAV                         SQZ        2563     A
SQZ                 LX6    IH.CAP-IH.RFP                                    SQZ        2564     A
SQZ                 SA1    "RN"+X6     RFW = RND(RF[R2W])                   SQZ        2565     A
SQZ                 BX6    X0*X6       RF[R2W] = 0                          SQZ        2566     A
SQZ                 SA3    ARFA                                             CCG0150       5     A
SQZ                 SX7    X1+         RF = RN[RFW]                         SQZ        2567     A
SQZ                 SB4    X7                                               CCG0150       6     A
SQZ                 BX3    X3*X1                                            CCG0150       7     A
SQZ                 LE     B4,B1,ARF   IF RF .LE. 1  */ NO RF               CCG0150       8     A
SQZ                 NZ     X3,ARF+1    ITERATE TO REMOVE REPEATED INCREMENTS CCG0150       9     A
SQZ                                                                         SQZ        2568     A
SQZ       ARF       ROUTINE                                                  SQZ        2569     A
SQZ                 BX2    X7                                       002603  CCG37         5     A
SQZ                 MX7    0                                        002604  CCG37         6     A
SQZ                 SA7    ARFAIL                                   002605  CCG37         7     A
SQZ                 BX7    X2                                       002606  CCG37         8     A
SQZ                 LX3    59-RN.CP                                         SQZ        2570     A
SQZ                 LX1    -RN.TPP                                          SQZ        2571     A
SQZ                 MI     X3,ARF1     IF C[RFW]                            SQZ        2572     A
SQZ                 LX3    RN.CP-RN.SAP                                     SQZ        2573          I
          -FCC0792
SQZ                 PL     X3,ARF3     IF ^SA[RFW]                          SQZ        2574          I
          -FCC0792
SQZ                 LX3    RN.CP-RN.IACP                                    FCC0792      20     A
SQZ                 MI     X3,ARF3     IF IAC[RFW]                          FCC0792      21     A
SQZ                 LX3    RN.IACP-RN.XMTP                                  FCC0792      22     A
SQZ                 MI     X3,ARF2     IF XMT[RFW]                          FCC0792      23     A
SQZ                                                                         SQZ        2575     A
SQZ       *         RF IS RESULT OF *  SA  RI,RJ*, SUBSTITUTE *RJ*          SQZ        2576     A
SQZ                                                                         SQZ        2577     A
SQZ                 SA3    X1          R1P = TP[RFW]                        SQZ        2578          I
          -FCC0792
SQZ                 SB4    OC.STT                                           SQZ        2579     A
SQZ                 NE     "OC",B4,ARF2      IF OC[R1] " OC.STT             SQZ        2580     A
SQZ                 BX2    X4                                               SQZ        2581     A
SQZ                 LX2    59-D.PRSP                                        SQZ        2582     A
SQZ                 MI     X2,ARF      IF PRS[DI]    */ AVOID B-REG DEADLOCKS SQZ        2583          I
          -FCC0448
SQZ                 PL     X2,ARF2     IF ^PRS[DI]   */ NO DEADLOCK POSSIBLE FCC0448     143     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                                                                                  FCC0448     144     A
SQZ                LX1    RN.TPP                                                     FCC0448     145     A
SQZ                EQ     ARF                                                        FCC0448     146     A
SQZ                                                                                  SQZ        2584     A
SQZ       ARF2     BX6    X0*X6       RF[R2W] = 0                                    SQZ        2585         I
         -FCC0792
SQZ       ARF2     SA3    X1          R1P = TP[RFW]                                  FCC0792      24     A
SQZ                BX6    X0*X6       RF[R2W] = 0                                    FCC0792      25     A
SQZ                LX3    -R1.RJP                                                    SQZ        2586     A
SQZ                BX7    -X0*X3      RF = RJ[R1P]                                   SQZ        2587     A
SQZ                SA1    "RN"+X7     RFW = RND(RF)                                  SQZ        2588     A
SQZ                EQ     ARF                                                        SQZ        2589     A
SQZ                                                                                  SQZ        2590     A
SQZ      *         *RF* IS *IAC*, ADD THE CONSTANT PART OF THE *RF* TO THE CA         SQZ        2591     A
SQZ                                                                                  SQZ        2592     A
SQZ       ARF3     SA3    X1          R1P = TP[R1W]                                  SQZ        2593     A
SQZ                SA1    X1+B2       DIP = R1P + 2                                  SQZ        2594     A
SQZ                LX3    -R1.RJP                                                    SQZ        2595     A
SQZ                BX7    -X0*X3                                                     SQZ        2596     A
SQZ                LX1    59-D.CMP                                                   SQZ        2597     A
SQZ                SA2    "RN"+X7     RNJ = RND(RJ[R1P])                             SQZ        2598     A
SQZ                LX3    R1.RJP-R1.RKP                                              SQZ        2599     A
SQZ                BX7    -X0*X3                                                     SQZ        2600     A
SQZ                SA3    "RN"+X7     RNK = RND(RK[R1P])                             SQZ        2601     A
SQZ                LX2    59-RN.CP                                                   SQZ        2602     A
SQZ                PL     X2,ARF4     IF ^C[RNJ]                                     SQZ        2603     A
SQZ                                                                                  SQZ        2604     A
SQZ                PL     X1,ARF7     IF ^CM[DIP]   */ IS                            SQZ        2605     A
SQZ                SX7    X3                                                         SQZ        2606     A
SQZ                LX2    1+RN.CP-RN.TPP                                             SQZ        2607     A
SQZ                BX6    X6+X7       RF[R2W] = RN[RNK]                              SQZ        2608     A
SQZ                SX3    X2                                                         SQZ        2609     A
SQZ                EQ     ARF5                                                       SQZ        2610     A
SQZ                                                                                  SQZ        2611     A
SQZ       ARF4     LX2    1+RN.CP                                                    SQZ        2612     A
SQZ                SX7    X2                                                         SQZ        2613     A
SQZ                BX6    X7+X6       RF[R2W] = RN[RNJ]                              SQZ        2614     A
SQZ                LX3    -RN.TPP                                                    SQZ        2615     A
SQZ                                                                                  SQZ        2616     A
SQZ       ARF5     SA3    X3          R1P = TP[RNJ]                                  SQZ        2617     A
SQZ                AX1    59          SGN = SHIFT( CM[DI] , -59 )                    SQZ        2618     A
SQZ                RJ     DCV                                                        SQZ        2619     A
SQZ                BX7    -X1-X7      SVAL = XOR( VAL[R1P] , ^SGN )                  SQZ        2620     A
SQZ                EQ     ARF0                                                       SQZ        2621     A
SQZ                                                                                  SQZ        2622     A
SQZ      *         CON - I CASE, LEAVE INSTRUCTION ALONE ( *LD* SQZ IN *PRE* )        SQZ        2623     A
SQZ                                                                                  SQZ        2624     A
SQZ       ARF7     SA3    A5+B1       R2 = R1 + 1                                    SQZ        2625     A
SQZ                LX3    -IH.RFP                                                    SQZ        2626     A
SQZ                SA1    "RN"+X3     RFW = RND(RF[R2])                              SQZ        2627     A
SQZ                BX6    X0*X3                                                      SQZ        2628     A
SQZ                MX7    1                                                 002608   CCG37         9     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                SA7    ARFAIL                                          002609   CCG37        10     A
SQZ                SX7    X1          RF = RN[RFW]                                 SQZ        2629     A
SQZ                EQ     ARF                                                      SQZ        2630     A
SQZ        EDD     TITLE  EDD - ELIMINATE DEAD DEFINITIONS                         SQZ        2631     A
SQZ       **       EDD - ELIMINATE DEAD DEFINITIONS ( USES = 0 ).                  SQZ        2632     A
SQZ       *        SCAN *TXT* BACKWARDS AND TRANSFER USES COUNTS TO DEFINITIONS.   SQZ        2633     A
SQZ       *        IF USES = 0 THEN MARK THE INSTRUCTION AS REDUNDANT.             SQZ        2634     A
SQZ                                                                                SQZ        2635     A
SQZ        EDD0    SA3    EDDA                                                     SQZ        2636     A
SQZ                BX6    X3                                                       SQZ        2637     A
SQZ                SA6    A5          RESTORE BOS                                  SQZ        2638     A
SQZ        EDD     ROUTINE                                                         SQZ        2639     A
SQZ                TRACE  EOQ,RLIST                                                CCG17       499     A
SQZ                SA3    "TXT"                                                    SQZ        2640     A
SQZ                MX7    0                                                        SQZ        2641          I
          -CCG17
SQZ                BX6    X3                                                       SQZ        2642     A
SQZ                SA6    EDDA                                                     SQZ        2643     A
SQZ                SA4    EOQA                                                     SQZ        2644          I
          -CCG17
SQZ                SA4    L.TXT                                                    CCG17       500     A
SQZ                SX5    "TXT"-4                                                  CCG17       501     A
SQZ                IX4    X4+X5                                                    CCG17       502     A
SQZ                MX7    0                                                        CCG17       503     A
SQZ                SA7    "TXT"       (O.TXT) = 0                                  SQZ        2645     A
SQZ                SA5    X4          TA = EOQA                                    SQZ        2646          I
          -CCG17
SQZ                SA5    X4          TA = TXT(L.TXT-4)  */ EOQ ADDRESS            CCG17       504     A
SQZ                SB4    OC.LDC                                                   CCG17       505     A
SQZ                SB5    59-D.^DP                                                 SQZ        2647     A
SQZ                SB6    60-IH.RFP                                                SQZ        2648     A
SQZ                MX1    -D.USESL                                                 SQZ        2649     A
SQZ                LX1    RN.USESP                                                 SQZ        2650     A
SQZ                SX2    B1                                                       SQZ        2651     A
SQZ                LX2    RN.USESP                                                 SQZ        2652     A
SQZ                EQ     EDD2                                                     SQZ        2653     A
SQZ                                                                                SQZ        2654     A
SQZ       *        TRANSFER USES COUNTS FROM *RND* TO INSTRUCTION DESCRIPTORS.     SQZ        2655     A
SQZ                                                                                SQZ        2656     A
SQZ        EDD1    LX6    D.USESP-RN.USESP                                         SQZ        2657     A
SQZ                BX7    X4+X6       USES[DI] = USES                              SQZ        2658     A
SQZ                SA7    A4                                                       SQZ        2659     A
SQZ                                                                                SQZ        2660     A
SQZ       *        SCAN *TXT* BACKWARDS FOR DEAD INSTRUCTIONS                      SQZ        2661     A
SQZ                                                                                SQZ        2662     A
SQZ        EDD2    SA4    A5-B2       DI = R1 - 2                                  SQZ        2663     A
SQZ                SA5    A4-B2       R1 = DI - 2                                  SQZ        2664     A
SQZ                ZR     X4,EDD2     IF [DI] = 0   */ DEAD INSTRUCTION            SQZ        2665     A
SQZ                LX6    B5,X4                                                    SQZ        2666     A
SQZ                BX7    -X0*X5                                                   SQZ        2667     A
SQZ                ZR     X5,EDD0     IF [R1] = 0   */ LOOP TERMINATOR             SQZ        2668     A
SQZ                MI     X6,EDD2     IF ^D[DI]     */ RI NOT A DEF                SQZ        2669     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                SA3    "RN"+X7                                                  SQZ        2670     A
SQZ                BX6    -X1*X3       USES = USES[RI]                             SQZ        2671     A
SQZ                NZ     X6,EDD1      IF USES " 0                                 SQZ        2672     A
SQZ                                                                                SQZ        2673     A
SQZ      *         USELESS DEFINITION.  MARK IT REDUNDANT AND DECREMENT USES       SQZ        2674     A
SQZ      *         OF ITS OPERANDS.                                                SQZ        2675     A
SQZ                                                                                SQZ        2676     A
SQZ                LX4    59-D.RJRSP                                               SQZ        2677     A
SQZ                MI     X4,EDD2      IF RJRS[DI]   */ PRECEDES LIVE RJ RS        SQZ        2678     A
SQZ                LX4    D.RJRSP-D.PRSP                                           SQZ        2679     A
SQZ                SA6    A4                                                       SQZ        2680     A
SQZ                PL     X4,EDD2A     IF ^PRS[DI]                                 SQZ        2681     A
SQZ                SA6    A4+4         [DI+4] = 0    */ KILL FOLLOWING REG STORE   SQZ        2682     A
SQZ       EDD2     LX4    1+D.PRSP+58-D.TYP [DI] = 0                               SQZ        2683     A
SQZ                IX7    X4+X4                                                    SQZ        2684     A
SQZ                MI     X7,EDD4      IF TYPE[DI] = II                            SQZ        2685     A
SQZ                MI     X4,EDD3      IF TYPE[DI] = III                           SQZ        2686     A
SQZ                                                                                SQZ        2687     A
SQZ      *         TYPE I                                                          SQZ        2688     A
SQZ                                                                                SQZ        2689     A
SQZ                LX5    -R1.RJP                                                  SQZ        2690     A
SQZ                BX6    -X0*X5                                                   SQZ        2691     A
SQZ                SA3    "RN"+X6                                                  SQZ        2692     A
SQZ                IX7    X3-X2        USES[RJ] = USES[RJ] - 1                     SQZ        2693     A
SQZ                SA7    A3                                                       SQZ        2694     A
SQZ                LX5    R1.RJP-R1.RKP                                            SQZ        2695     A
SQZ                BX6    -X0*X5                                                   SQZ        2696     A
SQZ                SA3    "RN"+X6                                                  SQZ        2697     A
SQZ                IX7    X3-X2        USES[RK] = USES[RK] - 1                     SQZ        2698     A
SQZ                SA7    A3                                                       SQZ        2699     A
SQZ                EQ     EDD2                                                     SQZ        2700     A
SQZ                                                                                SQZ        2701     A
SQZ      *         TYPE III                                                        SQZ        2702     A
SQZ                                                                                SQZ        2703     A
SQZ       EDD3     SA3    A5+B1        R2 = R1 + 1                                 SQZ        2704          I
         -CCG17
SQZ       EDD3     UX5    "OC",X5                                                  CCG17       506     A
SQZ                EQ     "OC",B4,EDD2 IF OC = LDC                                 CCG17       507     A
SQZ                SA3    A5+B1        R2 = R1 + 1                                 CCG17       508     A
SQZ                LX6    B6,X3                                                    SQZ        2705     A
SQZ                SA3    "RN"+X6                                                  SQZ        2706     A
SQZ                IX7    X3-X2        USES[RF] = USES[RF] - 1                     SQZ        2707     A
SQZ                SA7    A3                                                       SQZ        2708     A
SQZ                EQ     EDD2                                                     SQZ        2709     A
SQZ                                                                                SQZ        2710     A
SQZ      *         TYPE II , CHECK FOR AN RJ *RS*                                  SQZ        2711     A
SQZ                                                                                SQZ        2712     A
SQZ       EDD4     LX4    D.TYP+1-D.PIP                                            SQZ        2713     A
SQZ                PL     X4,EDD2      IF ^PI[DI]                                  SQZ        2714     A
SQZ                LX5    59-R1.INP                                                SQZ        2715     A
SQZ                PL     X5,EDD2      IF IN[R1] = 0 */ NOT A *RJRS*               SQZ        2716     A
SQZ                                                                                SQZ        2717     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ              SA3    A5-B2       DJP = R1 - 2                                  SQZ        2718    A
SQZ              SA4    A3-B2       R1J = DJP - 2                                 SQZ        2719    A
SQZ              SX5    B1                                                        SQZ        2720    A
SQZ              LX5    D.RJRSP                                                   SQZ        2721    A
SQZ              BX6    -X5*X3      RJRS[DJP] = 0                                 SQZ        2722    A
SQZ              LX4    -R1.RJP                                                   SQZ        2723    A
SQZ              SA6    A3                                                        SQZ        2724    A
SQZ              BX7    X0*X4       RJ[R1J] = 0                                   SQZ        2725    A
SQZ              LX7    R1.RJP                                                    SQZ        2726    A
SQZ              SA7    A4                                                        SQZ        2727    A
SQZ              EQ     EDD2                                                      SQZ        2728    A
SQZ                                                                              SQZ        2729    A
SQZ       EDDA   BSS    1                                                         SQZ        2730    A
SQZ       UCT    TITLE  UCT - UPDATE CVT/CUT TABLES                               CCG17       509    A
SQZ      **      UCT - UPDATE CUT/CVT TABLES TO ADD NEW CONSTANTS                 CCG17       510    A
SQZ                                                                              CCG17       511    A
SQZ       UCT    ROUTINE                                                          CCG17       512    A
SQZ              SA1    LDCN                                                      CCG17       513    A
SQZ              MI     X1,UCT      IF LDCN < 0   */ NO LDC"S ENCOUNTERED         CCG17       514    A
SQZ                                                                              CCG17       515    A
SQZ              ALLOC  CVT,X1      ALLOC( CVT,LDCN )                             CCG17       516    A
SQZ              SB5    X2                                                        CCG17       517    A
SQZ              SB7    B6          L = OLD(L.CVT)                                CCG17       518    A
SQZ              SB4    OC.LDC                                                    CCG17       519    A
SQZ              SA5    O.TXT                                                     CCG17       520    A
SQZ              SB2    B1+B1                                                     CCG17       521    A
SQZ              SA4    X5+B2       DI = TXT + 2                                  CCG17       522    A
SQZ              SA1    =XCON.      (X1) = CON.                                   CCG17       523         I
         -CCG18
SQZ              SA1    =XS$CON     (X1) = CON.                                   CCG18       163    A
SQZ              MX0    0           SD = 0                                        CCG17       524    A
SQZ                                                                              CCG17       525    A
SQZ      *       SCAN SEQUENCE FOR LIVE LDC"S, MOVE CONSTANT VALUE TO CVT,        CCG17       526    A
SQZ      *       ADJUST IH WORD                                                  CCG17       527    A
SQZ                                                                              CCG17       528    A
SQZ       UCT1   SA4    A4+4        DI = DI + 4                                   CCG17       529    A
SQZ              ZR     X4,UCT1     IF DI = 0     */ DEAD INSTRUCTION             CCG17       530    A
SQZ              SA5    A4-B2       R1 = DI - 2                                   CCG17       531    A
SQZ              UX7    "OC",X5                                                   CCG17       532    A
SQZ              ZR     "OC",UCT4   IF OC[R1] = 0 */ EOQ                          CCG17       533    A
SQZ              NE     "OC",B4,UCT1 IF OC[R1] " OC.LDC                           CCG17       534    A
SQZ                                                                              CCG17       535    A
SQZ              SA3    A4+B1       LI = DI + 1                                   CCG17       536    A
SQZ              BX0    X0+X4                                                     CCG17       537    A
SQZ              LX3    -I.CAP                                                    CCG17       538    A
SQZ              SX7    X3          C = CA[LI]                                    CCG17       539    A
SQZ              NZ     X7,UCT3     IF C " 0      */ LDC NOT GENERATED BY SQZ     CCG17       540    A
SQZ                                                                              CCG17       541    A
SQZ      *       SEARCH CVT FOR MATCH, ENTER IN TABLE                             CCG17       542    A
SQZ                                                                              CCG17       543    A
SQZ              SA3    A5+B1                                                     CCG17       544    A
SQZ              BX6    X3                                                        CCG17       545    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ              SA6    B5+B7       CVT(L) = [DI - 1]                          CCG17      546    A
SQZ              MX7    59          C = -1                                     CCG17      547    A
SQZ                                                                            CCG17      548    A
SQZ       UCT2   SX7    X7+1        C = C + 1                                  CCG17      549    A
SQZ              SA2    B5+X7       VALM = CVT(C)                              CCG17      550    A
SQZ              BX6    X3-X2                                                  CCG17      551    A
SQZ              NZ     X6,UCT2                                                CCG17      552    A
SQZ              MI     X6,UCT2     IF VAL " VALM */ NO MATCH                  CCG17      553    A
SQZ                                                                            CCG17      554    A
SQZ              SB3    X7                                                     CCG17      555    A
SQZ              LT     B3,B7,UCT3  IF C < L      */ MATCHING CVT ENTRY        CCG17      556    A
SQZ              SB7    B7+B1       L = L + 1                                  CCG17      557    A
SQZ                                                                            CCG17      558    A
SQZ       UCT3   LX7    IH.CAP                                                 CCG17      559    A
SQZ              BX6    X7+X1                                                  CCG17      560    A
SQZ              SA6    A4-1        [DI-1] = IH( 0,C,CON. )                    CCG17      561    A
SQZ              EQ     UCT1                                                   CCG17      562    A
SQZ                                                                            CCG17      563    A
SQZ       UCT4   LX0    59-D.PSP                                               CCG17      564           I
         -CCG20
SQZ              PL     X0,UCT5     */ NONE OF LDC"S PRECEDES A STORE INSTR.   CCG17      565           I
         -CCG20
SQZ              SX6    B1                                                     CCG17      566           I
         -CCG20
SQZ       UCT4   MX1    -D.STRSL                                               CCG20       12    A
SQZ              LX0    -D.STRSP                                               CCG20       13    A
SQZ              BX6    -X1*X0                                                 CCG20       14    A
SQZ              ZR     X6,UCT5     IF STRS[SD] = 0  */ NO PS ! PRS LDC"S      CCG20       15    A
SQZ              SA6    IXF                                                    CCG17      567    A
SQZ       UCT5   SX6    B7                                                     CCG17      568    A
SQZ              SA6    L.CVT       L.CVT = L                                  CCG17      569    A
SQZ              SX1    B7-B6       L = L - OLD(L.CVT)                         CCG17      570    A
SQZ              ZR     X1,UCT      IF NO NEW ENTRIES ADDED                    CCG17      571    A
SQZ              ALLOC  CUT,X1      ALLOC ( CUT,L )                            CCG17      572    A
SQZ              SETZERO X2+B6,X1                                              CCG17      573    A
SQZ              EQ     UCT                                                    CCG17      574    A
SQZ                                                                            CCG17      575    A
SQZ       EDI    TITLE  EDI - ELIMINATE DEAD INSTRUCTIONS                      SQZ       2731    A
SQZ      *       EDI - ELIMINATE DEAD INSTRUCTIONS ( COMPRESS *TXT* )          SQZ       2732    A
SQZ                                                                            SQZ       2733    A
SQZ       EDI0   SX6    B6-B5                                                  SQZ       2734    A
SQZ              SA6    L.TXT                                                  SQZ       2735    A
SQZ       EDI    ROUTINE                                                       SQZ       2736    A
SQZ              SA1    O.TXT                                                  CCG17      576    A
SQZ              SA2    O.RND                                                  CCG17      577    A
SQZ              S"TXT" X1                                                     CCG17      578    A
SQZ              S"RN"  X2                                                     CCG17      579    A
SQZ              MX0    -R1.RIL                                                CCG18      164    A
SQZ              MX7    0                                                      CCG9       127    A
SQZ              SA7    "TXT"+3     TXT(3) = 0    */ CLEAR LINK OF BOS         CCG9       128    A
SQZ              SA5    RSS                                                    SQZ       2737    A
SQZ              SB2    B1+B1                                                  CCG20       16    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                SB4    B2+B2                                                  SQZ        2738     A
SQZ                MX1    -D.USESL                                               SQZ        2739     A
SQZ                LX1    D.USESP                                                SQZ        2740     A
SQZ                SB3    -3                                                     SQZ        2741     A
SQZ                ZR     X5,EDI4     IF RSS = 0                                 SQZ        2742     A
SQZ                SB5    "TXT"                                                  CCG17       580     A
SQZ                                                                              SQZ        2743          I
          -CCG40
SQZ                SA4    "RN"                                                   CCG40        20     A
SQZ                LX4    -RN.TPP                                                CCG40        21     A
SQZ                SB6    X4                                                     CCG40        22     A
SQZ                SB6    B5-B6       BIAS = NEW(O.TXT)-OLD(O.TXT)               CCG40        23     A
SQZ      *         MOVE THE SEPARATED REGISTER STORES NEXT TO THEIR DEFINITIONS. SQZ        2744     A
SQZ                                                                              SQZ        2745     A
SQZ       EDI1     SA4    X5+B3       R1 = [RSS-3]                               SQZ        2746          I
          -CCG17
SQZ       EDI1     SA4    B5+X5       R1 = TXT + RSS                             CCG17       581     A
SQZ                SA3    A4+B2       DI = R1 + 2;  D = [DI]                     SQZ        2747     A
SQZ                MX7    0                                                      SQZ        2748     A
SQZ                SA7    A3          [DI] = 0                                   SQZ        2749     A
SQZ                BX6    -X1*X3                                                 SQZ        2750     A
SQZ                LX6    R1.INP+1-D.USESP                                       SQZ        2751     A
SQZ                BX2    -X0*X4                                                 SQZ        2752     A
SQZ                IX7    X4+X6       */ SAVE R1 OF RS & USES IN LINK WD OF DEF  SQZ        2753     A
SQZ                SA5    X5          RSS = [RSS]                                SQZ        2754          I
          -CCG17
SQZ                SA5    A4 - B3     RSS = [R1+3]                               CCG17       582          I
          -CCG18
SQZ                SA5    A4-B3       RSS = [RSS+3]                              CCG18       165     A
SQZ                SA2    "RN"+X2     RIA = O.RND + RI[R1]  */ RND WORD          SQZ        2755     A
SQZ                ZR     X3,EDI1A    IF D = 0      */ DEF IS DEAD               SQZ        2756     A
SQZ                LX2    -RN.TPP                                                SQZ        2757     A
SQZ                SA3    X2          R1P = TP[RIA]                              SQZ        2758          I
          -CCG40
SQZ                SA3    X2+B6       R1P = TP[R1A] + BIAS                       CCG40        24     A
SQZ                SA4    A4+B1       R2 = R1 + 1                                SQZ        2759     A
SQZ                BX6    -X3         [R1P] = -[R1P]  */ SEP RS FLAG             SQZ        2760     A
SQZ                SA6    A3                                                     SQZ        2761     A
SQZ                SA7    A6-B3       [R1P+3] = [R1] + USES[D]                   SQZ        2762     A
SQZ                BX6    X4                                                     SQZ        2763     A
SQZ                SA6    A2          [RIA] = [R2]  */ SAVE R2 OF RS IN *RND*    SQZ        2764     A
SQZ       EDI1A    NZ     X5,EDI1     IF RSS " 0                                 SQZ        2765          I
          -CCGA083
SQZ       EDI1A    SX5    X5                                                     CCGA083     533     A
SQZ                NZ     X5,EDI1     IF RSS .NE. 0                              CCGA083     534     A
SQZ                                                                              SQZ        2766     A
SQZ      *         NOW MOVE THE SEQUENCE UP                                       SQZ        2767     A
SQZ                                                                              SQZ        2768     A
SQZ                MX6    0                                                      SQZ        2769     A
SQZ                SA6    "TXT"       [O.TXT] = 0                                SQZ        2770     A
SQZ                MX1    -D.USESL                                               SQZ        2771     A
SQZ                LX1    R1.INP+1                                               SQZ        2772     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ              SA2    F.RDT+OC.RS                                          SQZ        2773    A
SQZ              SA3    EOQA                                                 SQZ        2774        I
         -CCG17
SQZ              SB5    X3-2        TA = EOQA - 2                            SQZ        2775        I
         -CCG17
SQZ              SA3    L.TXT                                                CCG17       583    A
SQZ              SB5    X3                                                   CCG17       584    A
SQZ              SX3    "TXT"+B5                                             CCG17       585    A
SQZ              SB5    X3-6        TA = EOQA - 2                            CCG17       586    A
SQZ              SA5    X3                                                   SQZ        2776        I
         -CCG18
SQZ              SB6    X3+B4       LWA + 1                                  SQZ        2777        I
         -CCG18
SQZ              SA5    X3-4                                                 CCG18       166    A
SQZ              SB6    X3          LWA+1                                    CCG18       167    A
SQZ              EQ     EDI3                                                 SQZ        2778    A
SQZ                                                                          SQZ        2779    A
SQZ       EDI2   SA3    A4+B1       R1RS = DI + 1                            SQZ        2780    A
SQZ              BX6    X1*X3                                                SQZ        2781    A
SQZ              BX3    -X1*X3                                               SQZ        2782    A
SQZ              SA6    B5-B2       [TA-2] = [R1RS]                          SQZ        2783    A
SQZ              LX3    D.USESP-R1.INP-1                                     SQZ        2784    A
SQZ              BX7    X2+X3       D = RDT(OC.RS) ! USES[R1RS]              SQZ        2785    A
SQZ              ZR     X3,EDI2A    IF USES[R1RS] = 0                        SQZ        2786    A
SQZ              SX3    B1                                                   SQZ        2787    A
SQZ              LX3    D.^DP                                                SQZ        2788    A
SQZ              BX7    -X3*X7      ^D[D] = 0                                SQZ        2789    A
SQZ       EDI2A  BX6    -X0*X6                                               SQZ        2790    A
SQZ              SA3    "RN"+X6     R2W = RND(RI[R1RS])                      SQZ        2791    A
SQZ              BX5    -X5         [R1] = -[R1]                             SQZ        2792    A
SQZ              BX6    X3                                                   SQZ        2793    A
SQZ              SA7    B5          [TA] = D                                 SQZ        2794    A
SQZ              SA6    B5-B1       [TA-1] = R2W                             SQZ        2795    A
SQZ              MX7    0                                                    SQZ        2796    A
SQZ              SA7    A7+B1       [TA+1] = 0                               SQZ        2797    A
SQZ              SB5    B5-B4       TA = TA - 4                              SQZ        2798    A
SQZ              EQ     EDI3A                                                SQZ        2799    A
SQZ                                                                          SQZ        2800    A
SQZ       EDI3   SA4    A5-B2       DI = R1 - 2                              SQZ        2801    A
SQZ              SA5    A4-B2       R1 = DI - 2                              SQZ        2802    A
SQZ              ZR     X4,EDI3     IF [DI] = 0                              SQZ        2803    A
SQZ       EDI3A  MI     X5,EDI2     IF [R1] < 0                              SQZ        2804    A
SQZ              SA3    A5+B1       R2 = R1 + 1                              SQZ        2805    A
SQZ              MX7    0                                                    SQZ        2806    A
SQZ              SA7    B5+B1       [R1+3] = 0                               SQZ        2807    A
SQZ              BX6    X4                                                   SQZ        2808    A
SQZ              SA6    B5                                                   SQZ        2809    A
SQZ              LX7    X3                                                   SQZ        2810    A
SQZ              SA7    B5-B1                                                SQZ        2811    A
SQZ              BX6    X5                                                   SQZ        2812    A
SQZ              SB5    B5-B4                                                SQZ        2813    A
SQZ              SA6    A6-B2                                                SQZ        2814    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                NZ     X5,EDI3     IF [R1] " 0                                  SQZ        2815     A
SQZ                                                                                SQZ        2816     A
SQZ                SA4    EDDA                                                     SQZ        2817     A
SQZ                SB5    A6          FWA                                          SQZ        2818     A
SQZ                SX6    A6                                                       SQZ        2819     A
SQZ                SA6    O.TXT                                                    SQZ        2820     A
SQZ                BX7    X4                                                       SQZ        2821     A
SQZ                SA7    B5                                                       SQZ        2822     A
SQZ                EQ     EDI0                                                     SQZ        2823     A
SQZ        EDI     EJECT                                                           SQZ        2824     A
SQZ       *        DEAD INSTRUCTION ELIMINATION WHEN NO SEPARATED REG STORES.      SQZ        2825     A
SQZ                                                                                SQZ        2826     A
SQZ        EDI4    SA4    "TXT"+B2    DI = TXT + 2                                 SQZ        2827     A
SQZ                SB5    "TXT"+6     TA = O.TXT + 6  */ STORE ADDRESS             SQZ        2828     A
SQZ                                                                                SQZ        2829     A
SQZ        EDI5    SA4    A4+B4       DI = DI + 4                                  SQZ        2830     A
SQZ                SA5    A4-B2       R1 = DI - 2                                  SQZ        2831     A
SQZ                ZR     X4,EDI5     IF [DI] = 0                                  SQZ        2832     A
SQZ                SA3    A4-B1       R2 = DI - 1                                  SQZ        2833     A
SQZ                MX7    0                                                        SQZ        2834     A
SQZ                BX6    X4                                                       SQZ        2835     A
SQZ                SA7    B5+B1       [R1+3] = 0                                   SQZ        2836     A
SQZ                UX2    "OC",X5                                                  SQZ        2837     A
SQZ                SA6    B5                                                       SQZ        2838     A
SQZ                BX7    X3                                                       SQZ        2839     A
SQZ                SA7    B5-B1                                                    SQZ        2840     A
SQZ                LX6    X5                                                       SQZ        2841     A
SQZ                SA6    B5-B2                                                    SQZ        2842     A
SQZ                SB5    B5+B4       TA = TA + 4                                  SQZ        2843     A
SQZ                NZ     "OC",EDI5   IF OC[R1] " OC.EOQ                           SQZ        2844     A
SQZ                SB6    A6+B4                                                    SQZ        2845     A
SQZ                SB5    "TXT"                                                    SQZ        2846     A
SQZ                EQ     EDI0                                                     SQZ        2847     A
SQZ        JUMPT   SPACE  2                                                        SQZ        2848          I
          -CCG17
SQZ        IXI     TITLE  IXI - INSERT EXTRA INSTRUCTIONS                          CCG17       587     A
SQZ                DESCRIBE L.                                                     CCG17       588     A
SQZ        OC      DEFINE 12          P(OC) XMT OR RS                              CCG17       589     A
SQZ        STU     DEFINE 12          N. ST USES                                   CCG17       590     A
SQZ        RST     DEFINE 18          R-NUM IN A STORE REGISTER                    CCG17       591     A
SQZ        RN      DEFINE 18          GENERAL R-NUMBER                             CCG17       592     A
SQZ                                                                                CCG17       593     A
SQZ       **       IXI - INSERT EXTRA INSTRUCTIONS FOR *LDC* OPCODES THAT *PS*     CCG17       594     A
SQZ       *        OR *S* THAT HAVE A NON-ZERO *SO* FIELD ( SEE *ECO4* CODE )      CCG17       595     A
SQZ       *        OR *IM*"S THAT NEED A UNPACK OF ONE OF ITS OPERANDS.            CCG17       596     A
SQZ                                                                                CCG17       597     A
SQZ        PRCS    MACRO  NAM                                                      CCG17       598     A
SQZ                BSS    0                                                        CCG17       599     A
SQZ        (NAM    EQU    *-IXI1                                                   CCG17       600     A
SQZ                ENDM                                                            CCG17       601     A
SQZ                                                                                CCG17       602     A
SQZ        IXI     ROUTINE                                                         CCG17       603     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                SA5    IXF                                                  CCG17        604     A
SQZ                ZR     X5,IXI      IF IXF = 0                               CCG17        605     A
SQZ                                                                            CCG17        606     A
SQZ                SX7    B1                                                   CCG17        607     A
SQZ                SB4    4                                                    CCG17        608     A
SQZ                LX7    58          N = 0                                    CCG17        609     A
SQZ                SX0    B4                                                   CCG17        610     A
SQZ                LX0    L.RSTP                                               CCG17        611     A
SQZ                SX6    B4                                                   CCG17        612     A
SQZ                IX0    X6+X0                                                CCG17        613     A
SQZ                SA1    O.TXT                                                CCG17        614     A
SQZ                MX2    -R1.SOL                                              CCG17        615     A
SQZ                LX2    R1.SOP                                               CCG17        616     A
SQZ                SA5    X1          R1 = O.TXT                               CCG17        617     A
SQZ                                                                            CCG17        618     A
SQZ      *         SCAN *TXT* FOR -                                            CCG17        619     A
SQZ      *                1) LDC INSTRUCTIONS THAT *PS*                        CCG17        620           I
         -CCG20
SQZ      *                1) LDC INSTRUCTIONS THAT *PS* OR *PRS* TO X0, X6 OR X7  CCG20      17     A
SQZ      *                2) S INSTRUCTIONS WITH *SO* " 0                      CCG17        621     A
SQZ      *                3) IM INSTRUCTIONS WITH THE *USI* BIT SET            CCG17        622     A
SQZ      *                4) DEF INSTRUCTIONS WIH NON-ZERO R2 WORD             CCGA083      535     A
SQZ                                                                            CCG17        623     A
SQZ       IXI1     SA5    A5+B4       R1 = R1 + 4                              CCG17        624     A
SQZ                SA4    A5+2        DI = R1 + 2                              CCG17        625     A
SQZ                UX6    "OC",X5                                              CCG17        626     A
SQZ                SA3    SBB.JT+"OC"                                          CCG17        627     A
SQZ                IX7    X7+X0       N = N + 4                                CCG17        628     A
SQZ                LX3    30                                                   CCG17        629     A
SQZ                SA7    A4+B1       LI = DI + 1;  [LI] = LW(0,0,N,N)         CCG17        630     A
SQZ                UX6    "OC",X3                                              CCG17        631     A
SQZ                JP     IXI1+"OC"   JUMP TO PROCESSOR                        CCG17        632     A
SQZ                                                                            CCG17        633     A
SQZ                PRCS   LDC                                                  CCG17        634     A
SQZ                LX4    59-D.PSP                                             CCG17        635     A
SQZ                PL     X4,IXI1     IF ^PS[DI]                               CCG17        636           I
         -CCG20
SQZ                SB7    P.XMT       PA = P.XMT    */ SET PROCESSOR ADDRESS   CCG17        637           I
         -CCG20
SQZ                MI     X4,IXI1A    IF PS[DI]                                CCG20         18     A
SQZ                LX4    D.PSP-D.PRSP                                         CCG20         19     A
SQZ                PL     X4,IXI1     IF ^PRS[DI]                              CCG20         20     A
SQZ                SA3    A4+2        R1RS = DI + 2                            CCG20         21     A
SQZ                MX5    -SO.RNL                                              CCG20         22     A
SQZ                LX3    -R1.SOP                                              CCG20         23     A
SQZ                BX5    -X5*X3                                               CCG20         24     A
SQZ                SB7    X5                                                   CCG20         25     A
SQZ                MX5    5                                                    CCG20         26     A
SQZ                LX5    -1                                                   CCG20         27     A
SQZ                LX5    B7,X5                                                CCG20         28     A
SQZ                MI     X5,IXI1     IF SORN[R1RS] IS 1 - 5                   CCG20         29     A
SQZ                                                                            CCG20         30     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ       IXI1A    SB7    P.XMT       PA = P.XMT    */ SET PROCESSOR ADDRESS          CCG20        31    A
SQZ                                                                                   CCG17       638    A
SQZ       IXI2     PX6    B7,X7                                                       CCG17       639    A
SQZ                SX4    B4                                                          CCG17       640    A
SQZ                LX4    L.RSTP                                                      CCG17       641    A
SQZ                IX6    X4+X6                                                       CCG17       642    A
SQZ                SA6    A4+B1       [LI] = LW(PA,0,N+4,N)                           CCG17       643    A
SQZ                IX7    X7+X0       N = N + 4                                       CCG17       644    A
SQZ                EQ     IXI1                                                        CCG17       645    A
SQZ                                                                                   CCG17       646    A
SQZ                PRCS   S                                                           CCG17       647    A
SQZ                BX3    -X2*X5                                                      CCG17       648    A
SQZ                ZR     X3,IXI1     IF SO[R1] = 0                                   CCG17       649    A
SQZ                SB7    P.RS        PA = P.RS                                       CCG17       650    A
SQZ                EQ     IXI2                                                        CCG17       651    A
SQZ                                                                                   CCGA083     536    A
SQZ                PRCS   DEF                                                         CCGA083     537    A
SQZ                SA3    A5+B1       R2 = R1 + 1                                     CCGA083     538    A
SQZ                LX5    B1,X3                                                       CCGA083     539    A
SQZ                PL     X5,IXI1     IF NOT BIT58[R2]  */ NOT BEF STORE              CCGA083     540    A
SQZ                SB7    P.ST                                                        CCGA083     541    A
SQZ                PX6    B7,X7                                                       CCGA083     542    A
SQZ                SA6    A4+B1       [LI] = LW(P.ST,0,N,N)                           CCGA083     543    A
SQZ                IX7    X7+X0       N = N + 4                                       CCGA083     544    A
SQZ                LX5    1                                                           CCGA083     545    A
SQZ                PL     X5,IXI1                                                     CCGA083     546    A
SQZ                IX7    X7+X0       N = N + 4                                       CCGA083     547    A
SQZ                EQ     IXI1                                                        CCGA083     548    A
SQZ                                                                                   CCG17       652    A
SQZ                PRCS   IM                                                          CCG17       653    A
SQZ                LX4    59-D.USIP                                                   CCG17       654    A
SQZ                PL     X4,IXI1     IF ^USI[DI]                                     CCG17       655    A
SQZ                IX7    X7+X0       N = N + 4                                       CCG17       656    A
SQZ                SB7    P.IM                                                        CCG17       657    A
SQZ                PX6    B7,X7                                                       CCG17       658    A
SQZ                SA6    A7          [LI] = LW(P.IM,0,N,N)                           CCG17       659    A
SQZ                EQ     IXI1                                                        CCG17       660    A
SQZ                                                                                   CCG17       661    A
SQZ                PRCS   EOQ                                                         CCG17       662    A
SQZ                SA2    L.TXT                                                       CCG17       663    A
SQZ                SX7    X7+B4       N = N + 4                                       CCG17       664    A
SQZ                IX1    X7-X2       NX = N - L.TXT                                  CCG17       665    A
SQZ                ALLOC  TXT,X1      ALLOC( TXT , NX )                               CCG17       666    A
SQZ                SB5    X3-4        SI = L.TXT - 4  */ STORE INDEX                  CCG17       667    A
SQZ                SB6    B6-4        FI = OLD(L.TXT)-4  */ FETCH INDEX               CCG17       668    A
SQZ                SB7    X2+3        TB = TXT + 3                                    CCG17       669    A
SQZ                ZR     X1,IXI20    IF NX = 0     */ NO EXTRA INSTRUCTIONS          CCG17       670    A
SQZ                TRACE  IXI,RLIST                                                   CCG17       671    A
SQZ                SX0    RN.MASK                                                     CCG17       672    A
SQZ                MX1    D.TYL                                                       CCG17       673    A
SQZ                SB5    B7+B5       SI = TB + SI                                    CCG17       674    A
SQZ                SB4    59-D.STP                                                    CCG17       675    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                                                                                  CCG17        676     A
SQZ      *         MOVE THE SEQUENCE UP IN *TXT* AND INSERT THE EXTRA INSTRUCTIONS   CCG17        677     A
SQZ                                                                                  CCG17        678     A
SQZ       IXI4     SA5    B7+B6       LI = TB + FI                                  CCG17        679     A
SQZ                UX4    B3,X5                                                     CCG17        680     A
SQZ                NZ     B3,IXI9     IF OC[LI] " 0                                 CCG17        681     A
SQZ                                                                                  CCG17        682     A
SQZ       IXI5     SA4    A5-B1       DI = LI - 1                                   CCG17        683     A
SQZ                SA3    A4-B1                                                     CCG17        684     A
SQZ                BX2    X3                                  */  R2 WORD           CCG0213        5     A
SQZ                BX7    X4                                                        CCG17        685     A
SQZ                LX3    -IH.RFP                                                   CCG17        686     A
SQZ                SA7    B5-B1       [SI-1] = [DI]                                 CCG17        687     A
SQZ                BX7    X2                                  */  R2 WORD           CCG0213        6     A
SQZ                BX2    -X1+X4                                                    CCG0213        7     A
SQZ                ZR     X2,IXI6     IF TY[DI] = IV                                CCG0213        8     A
SQZ                BX6    -X0*X3                                                    CCG17        688     A
SQZ                SA2    B7+X6       ADJ RF AND MOVE R2 WORD                       CCG17        689     A
SQZ                BX3    X0*X3                                                     CCG17        690     A
SQZ                SX2    X2                                                        CCG17        691     A
SQZ                IX7    X2+X3                                                     CCG17        692     A
SQZ                LX7    IH.RFP                                                    CCG17        693     A
SQZ                SA7    A7-B1                                                     CCG17        694           I
          -CCG0213
SQZ                                                                                 CCG0213        9     A
SQZ       IXI6     SA7    A7-B1                                                     CCG0213       10     A
SQZ                                                                                  CCG17        695     A
SQZ                SA3    A3-B1       ADJ R1 WORD                                   CCG17        696     A
SQZ                BX2    -X1+X4                                                    CCG17        697     A
SQZ                ZR     X2,IXI7     IF TY[DI] = IV                                CCG17        698     A
SQZ                BX5    X1*X4                                                     CCG17        699     A
SQZ                BX6    -X0*X3                                                    CCG17        700     A
SQZ                SA2    B7+X6       LII = [TB+RI[R1]]                             CCG17        701     A
SQZ                BX3    X0*X3                                                     CCG17        702     A
SQZ                LX6    B4,X4                                                     CCG17        703     A
SQZ                PL     X6,IXI7     IF ^ST[DI]                                    CCG17        704     A
SQZ                IFNE   .DAL,0,2                                                  CCG17        705     A
SQZ                LX6    D.STP-D.L2P                                               CCG17        706     A
SQZ                MI     X6,IXI7     IF L2[DI]                                     CCG17        707     A
SQZ                                                                                  CCG17        708     A
SQZ                SX6    B1                                                        CCG17        709     A
SQZ                LX6    L.STUP                                                    CCG17        710     A
SQZ                IX6    X2+X6       STU[LII] = STU[LII] + 1                       CCG17        711     A
SQZ                SA6    A2                                                        CCG17        712     A
SQZ                LX2    -L.RSTP                                                   CCG17        713     A
SQZ                                                                                  CCG17        714     A
SQZ       IXI7     SX2    X2                                                        CCG17        715     A
SQZ                BX3    X3+X2                                                     CCG17        716     A
SQZ                NZ     X5,IXI8     IF TY[DI] " I                                 CCG17        717     A
SQZ                                                                                  CCG17        718     A
SQZ                LX3    -R1.RJP                                                   CCG17        719     A
SQZ                BX6    -X0*X3                                                    CCG17        720     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                SA2    B7+X6                                                  CCG17        721     A
SQZ                BX3    X0*X3                                                  CCG17        722     A
SQZ                SX2    X2                                                     CCG17        723     A
SQZ                BX3    X2+X3                                                  CCG17        724     A
SQZ                LX3    R1.RJP-R1.RKP                                          CCG17        725     A
SQZ                BX6    -X0*X3                                                 CCG17        726     A
SQZ                SA2    B7+X6                                                  CCG17        727     A
SQZ                BX3    X0*X3                                                  CCG17        728     A
SQZ                SX2    X2                                                     CCG17        729     A
SQZ                BX3    X2+X3                                                  CCG17        730     A
SQZ                LX3    R1.RKP                                                 CCG17        731     A
SQZ                LX4    59-D.RJRSP                                             CCG17        732     A
SQZ                PL     X4,IXI8     IF ^RJRS[DI]                               CCG17        733     A
SQZ                                                                             CCG17        734     A
SQZ      *         CORRECT THE VALUE OF THE *RJ* FIELD SINCE THE BACKWARD       CCG17        735     A
SQZ      *         PROCESSSING OF THE SEQUENCE HAS ALREADY CLEARED THE LINK WORD CCG17       736     A
SQZ                                                                             CCG17        737     A
SQZ                BX7    -X0*X3                                                 CCG17        738     A
SQZ                SX7    X7+4                                                   CCG17        739     A
SQZ                LX7    R1.RJP      RJ[R1] = RI[R1] + 4                        CCG17        740     A
SQZ                BX3    X7+X3                                                  CCG17        741     A
SQZ                                                                             CCG17        742     A
SQZ       IXI8     BX7    X3                                                     CCG17        743     A
SQZ                SA7    A7-B1                                                  CCG17        744     A
SQZ                MX6    0                                                      CCG17        745     A
SQZ                SA6    B5          [SI] = 0                                   CCG17        746     A
SQZ                SB5    B5-4        SI = SI - 4                                CCG17        747     A
SQZ                SB6    B6-4        FI = FI - 4                                CCG17        748     A
SQZ                GT     B6,IXI4     IF FI > 0                                  CCG17        749     A
SQZ                                                                             CCG17        750     A
SQZ                TRACE  IXIX,RLIST                                             CCG17        751     A
SQZ                EQ     IXI                                                    CCG17        752     A
SQZ                                                                             CCG17        753     A
SQZ      *         ADD AN INSTRUCTION AFTER/BEFORE THE CURRENT INSTRUCTION      CCG17        754     A
SQZ                                                                             CCG17        755     A
SQZ                PURGMAC PRCS                                                 CCG17        756     A
SQZ       PRCS     MACRO  NAM                                                    CCG17        757     A
SQZ                BSS    0                                                      CCG17        758     A
SQZ       P.NAM    EQU    *-IXI9                                                 CCG17        759     A
SQZ                ENDM                                                         CCG17        760     A
SQZ                                                                             CCG17        761     A
SQZ       IXI9     MX7    0                                                      CCG17        762     A
SQZ                SA7    A5          [LI] = 0                                   CCG17        763         I
          -CCG20
SQZ                SA7    B5          [LI] = 0                                   CCG20         32     A
SQZ                JP     IXI9+B3                                                CCG17        764     A
SQZ                                                                             CCG17        765     A
SQZ      *         ADD AN *XMT* AFTER A *LDC* THAT *PS*                         CCG17        766         I
          -CCG41
SQZ      *         ADD AN *XMT* AFTER AN *LDC* THAT *PS* OR *PRS*       002450  CCG41         12     A
SQZ                                                                             CCG17        767     A
SQZ                PRCS   XMT                                                    CCG17        768     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ          SA3    F.RDT+OC.XMT                                                  CCG17     769    A
SQZ          SB3    OC.XMT                                                        CCG17     770    A
SQZ          SA4    A5-B1       DI = LI - 1                                       CCG17     771    A
SQZ          SX2    B1                                                            CCG17     772         I
        -CCG41
SQZ          BX6    X4                                                   002470   CCG41      13    A
SQZ          LX6    59-D.PSP                                             002480   CCG41      14    A
SQZ          MI     X6,IXI10          IF PS[DI]                          002490   CCG41      15    A
SQZ          SA2    B5+B1       R1RS = SI + 1                            002500   CCG41      16    A
SQZ          SX7    4                                                    002510   CCG41      17    A
SQZ          IX6    X2+X7       RI[R1RS] = RI[R1RS] + 4                  002520   CCG41      18    A
SQZ          SA6    A2                                                   002530   CCG41      19    A
SQZ          SX2    B1                                                   002540   CCG41      20    A
SQZ          LX6    -R1.SOP-SO.LKP                                       002550   CCG41      21    A
SQZ          BX6    -X7*X2      B = ^SOLK[R1RS] & 1                      002560   CCG41      22    A
SQZ          LX2    D.PRSP                                               002570   CCG41      23    A
SQZ          BX3    X3+X2       PRS[DI] = 1                              002580   CCG41      24    A
SQZ          BX7    X3+X6       USES[XMT] = B */ =1 IF UJP/FULL LOCK     002590   CCG41      25    A
SQZ          SA7    B5-B1                                                002600   CCG41      26    A
SQZ          BX4    -X2*X4      PRS[DI] = 0                              002610   CCG41      27    A
SQZ          IX6    X4-X6       USES[DI] = USES[DI] - B                  002620   CCG41      28    A
SQZ          SA6    A4                                                   002630   CCG41      29    A
SQZ          LX5    -L.STUP                                              002640   CCG41      30    A
SQZ          EQ     IXI11                                                002650   CCG41      31    A
SQZ                                                                      002660   CCG41      32    A
SQZ     IXI10  SX2    B1                                                  002670   CCG41      33    A
SQZ          IX4    X4+X2       USES[DI] = USES[DI] + 1                           CCG17     773    A
SQZ          LX2    D.PSP                                                         CCG17     774    A
SQZ          BX3    X3+X2       PS[XMT] = 1                                       CCG17     775    A
SQZ          BX4    -X2*X4      PS[DI] = 0                                        CCG17     776    A
SQZ          LX5    -L.STUP                                                       CCG17     777    A
SQZ          MX6    -D.USESL                                                      CCG17     778    A
SQZ          BX6    -X6*X5                                                        CCG17     779    A
SQZ          BX7    X3+X6       USES[XMT] = STU[LI]                               CCG17     780    A
SQZ          SA7    B5-B1                                                         CCG17     781    A
SQZ          IX6    X4-X6       USES[DI] = USES[DI] - STU[LI]                     CCG17     782    A
SQZ          SA6    A4                                                            CCG17     783    A
SQZ          MX7    0                                                             CCG17     784         I
        -CCG41
SQZ                                                                      002690   CCG41      34    A
SQZ     IXI11  MX7    0                                                   002700   CCG41      35    A
SQZ          SA7    A7-B1                                                         CCG17     785    A
SQZ          LX5    L.STUP-L.RSTP                                                 CCG17     786    A
SQZ          SX6    X5                                                            CCG17     787    A
SQZ          LX5    L.RSTP                                                        CCG17     788    A
SQZ          SX5    X5                                                            CCG17     789    A
SQZ          PX6    B3,X6                                                         CCG17     790    A
SQZ          LX5    R1.RJP                                                        CCG17     791    A
SQZ          BX7    X6+X5                                                         CCG17     792    A
SQZ          SA7    A7-B1       TYI(OC.XMT,0,RN,SRN)                              CCG17     793    A
SQZ          SB5    B5-4        SI = SI - 4                                       CCG17     794    A
SQZ          EQ     IXI5                                                          CCG17     795    A

        0         1         2         3         4         5         6         7         8
        12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                                                                                  CCGA083     549     A
SQZ     *         ADD A ST TO ST. AFTER DEF ( RJ3 SQZ )                              CCGA083     550     A
SQZ                                                                                  CCGA083     551     A
SQZ               PRCS   ST                                                          CCGA083     552     A
SQZ               SB2    A5                                                          CCGA083     553     A
SQZ               RJ     IFT         INSERT FIRST STORE                              CCGA083     554     A
SQZ               SA2    A5-7                                                        CCGA083     555     A
SQZ               UX2    B3,X2                                                       CCGA083     556     A
SQZ               SX2    B3-OC.DEF                                                   CCGA083     557     A
SQZ               NZ     X2,IXI5     IF NOT DBL PREC                                 CCGA083     558     A
SQZ               SB2    A5-4                                                        CCGA083     559     A
SQZ               RJ     IFT                                                         CCGA083     560     A
SQZ               EQ     IXI5                                                        CCGA083     561     A
SQZ                                                                                  CCGA083     562     A
SQZ       IFT     ROUTINE                                                            CCGA083     563     A
SQZ               SA3    F$RDT+OC.ST                                                 CCGA083     564     A
SQZ               SB3    OC.ST                                                       CCGA083     565     A
SQZ               BX6    X3                                                          CCGA083     566     A
SQZ               SA3    B2-2                                                        CCGA083     567     A
SQZ               SA6    B5-B1       DESCR                                           CCGA083     568     A
SQZ               UX6    X3                                                          CCGA083     569     A
SQZ               SA6    A6-B1       R2W = CAIH[R2DEF]                               CCGA083     570     A
SQZ               SA3    B2          LINK WORD OF DEF                                CCGA083     571     A
SQZ               BX6    -X0*X3                                                      CCGA083     572     A
SQZ               PX6    B3,X6                                                       CCGA083     573     A
SQZ               SA6    A6-B1       R1W = TYIII(OC.ST,0,0,RI[R1DEF])                CCGA083     574     A
SQZ               MX6    0                                                           CCGA083     575     A
SQZ               SA6    B5          CLEAR LINK OF ST                                CCGA083     576     A
SQZ               SA6    B2-2        R2DEF = 0                                       CCGA083     577     A
SQZ               SB5    B5-4        SI = SI - 4                                     CCGA083     578     A
SQZ               EQ     IFT                                                         CCGA083     579     A
SQZ                                                                                  CCG17       796     A
SQZ     *         ADD AN *RS* AFTER A *S* WITH *SO* " 0                              CCG17       797     A
SQZ                                                                                  CCG17       798     A
SQZ               PRCS   RS                                                          CCG17       799     A
SQZ               SA3    F.RDT+OC.RS                                                 CCG17       800     A
SQZ               SB3    OC.RS                                                       CCG17       801     A
SQZ               BX6    X3                                                          CCG17       802     A
SQZ               SA6    B5-B1                                                       CCG17       803     A
SQZ               SA7    A6-B1                                                       CCG17       804     A
SQZ               SX6    X5                                                          CCG17       805     A
SQZ               SA3    A5-3        R1 = LI - 3                                     CCG17       806     A
SQZ               PX7    B3,X6                                                       CCG17       807     A
SQZ               MX6    -R1.SOL                                                     CCG17       808     A
SQZ               LX6    R1.SOP                                                      CCG17       809     A
SQZ               BX5    -X6*X3                                                      CCG17       810     A
SQZ               BX7    X5+X7                                                       CCG17       811     A
SQZ               SA7    A7-B1       TYII(OC.RS,0,SO,RI)                             CCG17       812     A
SQZ               BX6    X6*X3                                                       CCG17       813     A
SQZ               SA6    A3          CLEAR SO FIELD OF *S*                           CCG17       814     A
SQZ               SB5    B5-4                                                        CCG17       815     A
SQZ               EQ     IXI5                                                        CCG17       816     A

                  0         1         2         3         4         5         6         7         8
                  12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                                                                    CCG17     817   A
SQZ     *         *IM* WITH AN UNSAFE OPERAND, ADD A *UP* BEFORE       CCG17     818   A
SQZ                                                                    CCG17     819   A
SQZ               PRCS   IM                                            CCG17     820   A
SQZ               SA3    F.RDT+OC.IM                                   CCG17     821   A
SQZ               SB3    OC.IM                                         CCG17     822   A
SQZ               SA3    A5-B1       DESCR OF IM                       CCG17     823   A
SQZ               SX2    B1                                            CCG17     824   A
SQZ               LX2    D.USIP                                        CCG17     825   A
SQZ               BX6    -X2*X3      CLEAR USI BIT IN *IM*             CCG17     826   A
SQZ               SA6    B5-B1                                         CCG17     827   A
SQZ               SA7    A6-B1       R2 = 0                            CCG17     828   A
SQZ               SX7    X5          I = RN[LI]    */ NEW RI           CCG17     829   A
SQZ               SA5    A5-3        R1 = LI - 3                       CCG17     830   A
SQZ               LX5    -R1.RJP                                       CCG17     831   A
SQZ               BX2    -X0*X5      J = RJ[R1]                        CCG17     832   A
SQZ               LX5    R1.RJP-R1.RKP                                 CCG17     833   A
SQZ               BX3    -X0*X5                                        CCG17     834   A
SQZ               SA4    B7+X2                                         CCG17     835   A
SQZ               SA5    B7+X3                                         CCG17     836   A
SQZ               SX4    X4          NJ = RN[TB(J)]                    CCG17     837   A
SQZ               SX5    X5          NK = RN[TB(K)]                    CCG17     838   A
SQZ               SX3    X7                (X3) = I                    CCG17     839   A
SQZ               SA2    A4-B1       DJ = LJ - 1                       CCG17     840   A
SQZ               CALL   CMO                                           CCG17     841   A
SQZ               ZR     X6,IXI12    IF J IS SAFE                      CCG17     842   A
SQZ               SX2    X4          UK = NJ                           CCG17     843   A
SQZ               SX4    X3-4        NJ = I - 4                        CCG17     844   A
SQZ               EQ     IXI13                                         CCG17     845   A
SQZ                                                                    CCG17     846   A
SQZ       IXI12   SX2    X5          UK = NK                           CCG17     847   A
SQZ               SX5    X3-4        NK = I - 4                        CCG17     848   A
SQZ                                                                    CCG17     849   A
SQZ       IXI13   PX6    B3,X3                                         CCG17     850   A
SQZ               LX4    R1.RJP                                        CCG17     851   A
SQZ               LX5    R1.RKP                                        CCG17     852   A
SQZ               BX4    X4+X5                                         CCG17     853   A
SQZ               BX6    X6+X4                                         CCG17     854   A
SQZ               SA6    A7-B1       NEW R1 = TYI(OC.IM,I,NJ,NK)       CCG17     855   A
SQZ               SB3    OC.UP                                         CCG17     856   A
SQZ               SX5    X3-4                                          CCG17     857   A
SQZ               SA4    F.RDT+B3                                      CCG17     858   A
SQZ               MX6    0                                             CCG17     859   A
SQZ               SA6    A6-B1       LINK WD = 0                       CCG17     860   A
SQZ               SX7    B1                                            CCG17     861   A
SQZ               BX7    X7+X4                                         CCG17     862   A
SQZ               SA7    A6-B1             DESCR + USES                CCG17     863   A
SQZ               SA6    A7-B1       R2 =  0                           CCG17     864   A
SQZ               PX7    B3,X5                                         CCG17     865   A
SQZ               LX2    R1.RKP                                        CCG17     866   A
SQZ               BX6    X7+X2                                         CCG17     867   A
SQZ               SA6    A6-B1       R1 UP = TYI(OC.UP,I-4,0,UK)       CCG17     868   A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                SB5    B5-8        SI = SI - 8                                  CCG17        869     A
SQZ                SB6    B6-4        FI = FI - 4                                  CCG17        870     A
SQZ                EQ     IXI4                                                     CCG17        871     A
SQZ                                                                                CCG17        872     A
SQZ       *        NO INSTRUCTIONS TO BE MODIFIED, CLEAR LINK WORDS                CCG17        873     A
SQZ                                                                                CCG17        874     A
SQZ        IXI20   MX7    0                                                        CCG17        875     A
SQZ                SB3    B5+B7                                                    CCG17        876     A
SQZ                SB4    4                                                        CCG17        877     A
SQZ                                                                                CCG17        878     A
SQZ        IXI21   SA7    B7          [LI] = 0                                     CCG17        879     A
SQZ                SB7    B7+B4       LI = LI + 4                                  CCG17        880     A
SQZ                LE     B7,B3,IXI21                                              CCG17        881     A
SQZ                EQ     IXI                                                      CCG17        882     A
SQZ        CMO     SPACE  3,14                                                     CCG17        883     A
SQZ       **       CMO - CHECK MULTIPLY ( IM ) OPERAND TO SEE IF IT IS "SAFE"      CCG17        884     A
SQZ       *        SO THAT THE *IM* CAN BE MOVED FROM THE SIDE BRANCH OF A LOOP.   CCG17        885     A
SQZ       *                                                                        CCG17        886     A
SQZ       *        ENTRY  (A2,X2) = D, DESCRIPTOR WORD                             CCG17        887     A
SQZ       *                                                                        CCG17        888     A
SQZ       *        EXIT   (X6) = 0 IF OPERAND IS SAFE                              CCG17        889     A
SQZ       *                                                                        CCG17        890     A
SQZ       *        USES   A2, B2, X2, X6, X7                                       CCG17        891     A
SQZ                                                                                CCG17        892     A
SQZ        CMO     ROUTINE                                                         CCG17        893     A
SQZ                LX2    59-D.ZPP                                                 CCG17        894     A
SQZ                MX6    1           CMO = 1S59    */ UNSAFE                      CCG17        895     A
SQZ                PL     X2,CMO      IF ^ZP[D]     */ UNKNOWN OPERAND             CCG17        896     A
SQZ                SA2    A2-2        R1 = D - 2                                   CCG17        897     A
SQZ                UX7    B2,X2                                                    CCG17        898     A
SQZ                MX6    0           CMO = 0                                      CCG17        899     A
SQZ                SX7    B2-OC.LD                                                 CCG17        900     A
SQZ                NZ     X7,CMO      IF OC[R1] " OC.LD */ FMA OR S                CCG17        901     A
SQZ         .VD    IF     -DEF,VD.MATP                                             CCG18        168     A
SQZ                MX6    1                                                        CCG18        169     A
SQZ         .VD    ELSE                                                            CCG18        170     A
SQZ                SA2    A2+B1       R2 = R1 + 1                                  CCG17        902     A
SQZ                SX7    X2                                                       CCG17        903     A
SQZ                SA2    =XVD.                                                    CCG17        904         I
          -CCG18
SQZ                SA2    =XS$VD                                                   CCG18        171     A
SQZ                BX6    X2-X7       CMO = IH[R2] - VD.                           CCG17        905     A
SQZ         .VD    ENDIF                                                           CCG18        172     A
SQZ                EQ     CMO                                                      CCG17        906     A
SQZ        JUMPT   EJECT                                                           CCG17        907     A
SQZ       **       OPR - DEFINE MACRO TO FORM JUMP TABLE FOR OPCODES               SQZ         2849     A
SQZ                                                                                SQZ         2850     A
SQZ                MACRO  OPR,NAM                                                  SQZ         2851     A
SQZ                IF     DEF,.NAM,2                                               SQZ         2852         I
          -CCG17
SQZ                EQ     .NAM                                                     SQZ         2853         I
          -CCG17

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

```
SQZ                ELSE   1                                                  SQZ        2854          I
           -CCG17
SQZ                EQ     SBB+4S15                                           SQZ        2855          I
           -CCG17
SQZ                ENDM                                                      SQZ        2856          I
           -CCG17
SQZ                IF     -DEF,.NAM,2                                        CCG17       908     A
SQZ                EQ     *+1S17                                             CCG17       909     A
SQZ                SKIP   9                                                  CCG17       910     A
SQZ      *                         SBB OP CODE JUMP                         CCG17       911     A
SQZ      +         VFD    12/0400B,18/.NAM                                   CCG17       912     A
SQZ      *                         IXI RELATIVE JUMP ADDRESS                CCG17       913     A
SQZ                IF     DEF,(NAM,2                                         CCG17       914     A
SQZ                VFD    12/1S10+(NAM                                       CCG17       915     A
SQZ                SKIP   1                                                  CCG17       916     A
SQZ                VFD    12/1S10                                            CCG17       917     A
SQZ      *                         SBB JUMP FOR CONSTANT OPERANDS CASE      CCG17       918     A
SQZ                IF     DEF,)NAM,2                                         CCG17       919     A
SQZ                VFD    18/)NAM                                            CCG17       920     A
SQZ                SKIP   1                                                  CCG17       921     A
SQZ                VFD    18/.NAM                                            CCG17       922     A
SQZ                ENDM                                                      CCG17       923     A
SQZ                                                                          CCG17       924     A
SQZ                                                                          SQZ        2857     A
SQZ       SBB.JT   BSS    0                                                  SQZ        2858     A
SQZ                LOC    0                                                  SQZ        2859     A
SQZ      *CALL     OPRDEFS                                                   SQZ        2860     A
SQZ                LOC    *O                                                 SQZ        2861     A
SQZ                LIST   -G                                                 SQZ        2862     A
SQZ                                                                          SQZ        2863     A
SQZ                QUAL                                                      SQZ        2864     A
SQZ       CMO      EQENT  /SQUEEZE/CMO                                       CCG17       925     A
SQZ       SBB      EQU    /SQUEEZE/SBB                                       SQZ        2865          I
           -CCGNC
SQZ                                                                          CCGNC       765     A
SQZ                ECHO   2,Z=(SBB,SIE,SQZB)                                 CCGNC       766     A
SQZ                ENTRY  ;A#                                               CCGNC       767     A
SQZ       ;A#      EQU    /SQUEEZE/;A                                       CCGNC       768     A
SQZ                                                                          CCGNC       769     A
SQZ                END                                                       SQZ        2866     A
```

SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - SQZ

IDENTIFIER	TOTAL	ACTIVE
SQZ	2866	2384
FCC0448	4	4
FCC0509	21	19
FCC0514	87	83
FCC0588	9	9
FCC0603	5	5
FCC0643	2	2

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

FCC0716	39	39
FCC0715	7	7
FCC0792	22	22
FCC0872	3	3
FCC1025	27	27
FCC0957	14	14
FCC1025A	2	2
FCC1029	6	6
FCC1085	22	22
FCC1413	11	11
FCC1424	2	1
CCG1	22	6
CCG2	18	18
CCG3	116	115
CCG7	5	3
CCG9	3	3
FCC1660X	26	26
FCC1719X	5	5
CCGNC	24	23
CCG14	3	2
FCC1825X	20	20
CCG16	7	7
CCG17	925	886
CCG18	27	25
FCC1928	6	6
CCG20	32	31
CCG21	6	4
CCG22	84	78
CCG24	3	3
CCG27	62	61
CCG28	6	6
FCC2072	15	15
CCG34	5	5
CCG35	2	2
CCG37	9	9
CCG40	13	13
CCG41	24	24
CCGA013	1	1
CCGA024	17	17
CCGA039	9	9
CCGA045	2	2
CCGA047	5	5
CCGA049	7	7
CCG0006	8	8
CCG0002	13	13
CCG0041	7	7
CCGA083	574	565
CCG0012	2	2
CCG0124	1	1
CCGA085	5	5
CCG0147	3	3
CCG0150	5	5

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN SQZ

                CCG0154                    7         7
                CCG0158                    5         5
                CCG0181                    2         0
                CCGA094                    5         5
                CCG0189                    2         2
                CCG0193                    8         8
                CCG0206                    4         4
                CCG0211                   24        24
                CCG0213                    6         6
                CCG0216                   11        11

                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

      MCG       *DECK     MCG                                                  MCG          1     A
      MCG                 IDENT  MCG                                           MCG          2     A
      MCG        MCG      TITLE  MCG - MACHINE CODE GENERATOR                  MCG          3     A
      MCG       *CALL     SSTCALL                                              MCG          4     A
      MCG                 PASS2TM                                              MCG          5           I
                -CCG1
      MCG        B=MCG    RPVDEF                                               MCG          6     A
      MCG                                                                      MCG          7     A
      MCG                 TABLES TXT,TREE,PIT,OTI                              MCG          8           I
                -CCG1
      MCG                 ECHO   1,Z=(OTI,PIT,TREE,TXT)                        CCG1       281           I
                -CCGNC
      MCG                 EXT    O.Z,L.Z                                       CCG1       282           I
                -CCGNC
      MCG                 TABLES (OTI,PIT,TREE,TXT)                            CCGNC      770     A
      MCG       *IF       DEF,MCG,1                                            MCG          9     A
      MCG                 TRACER (II,FINAL,HEAD)                               MCG         10           I
                -CCG1
      MCG                 TRACER (JAM,HEAD,FINAL,II)                           CCG1       283     A
      MCG                                                                      MCG         11     A
      MCG        INCT     MICRO  .CPU-71,1,/11411/ INCREMENT UNIT TIME         MCG         12     A
      MCG        MCG      SPACE  3                                             MCG         13     A
      MCG       **        MCG - MACHINE CODE GENERATOR - SCHEDULING AND LOCAL REGISTER  MCG  14     A
      MCG       *         ASSIGNMENT.                                          MCG         15     A
      MCG       *                                                              MCG         16     A
      MCG       *         S.I. JASIK - MARCH 73                                MCG         17     A
      MCG       *                                                              MCG         18     A
      MCG       *         *MCG* ORDERS THE INSTRUCTIONS IN THE CODE SEQUENCE SUBJECT TO  MCG  19     A
      MCG       *         THE CONSTRAINTS OF THE PARTIAL ORDER IMPOSED ON IT BY *BDT*,   MCG  20     A
      MCG       *         WHICH ARE -                                          MCG         21     A
      MCG       *          1) DATA INTERFERENCE DEPENDENCIES ( LD/ST INTERFERENCE )  MCG  22     A
      MCG       *          2) OPERAND DEPENDENCIES                             MCG         23     A
      MCG       *          3) BOUNDARY MARKER DEPENDENCIES  ( AN INSTRUCTION MUST BE  MCG  24     A
      MCG       *         LINKED TO THE NODES OF THE BOUNDARIES OF THE BASIC BLOCK THAT  MCG  25     A
      MCG       *         IT BELONGS TO ).                                     MCG         26     A
      MCG       *          4) EXPLICIT USE OF SPECIFIC RESOURCES ( REGISTERS ).  MCG  27     A
      MCG       *            I.E. THE REUSE/REDEFINITION OF A REGISTER MAY NOT OCCUR  MCG  28     A
      MCG       *         UNTIL ALL USES OF THE FIRST DEFINITION HAVE BEEN EXHAUSTED.  MCG  29     A
      MCG       *                                                              MCG         30     A

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG     *        THE RESULTANT NETWORK/DEPENDENCY GRAPH ( NOT A TREE ) IS          MCG      31    A
MCG     *        EXPRESSED AS A SUCCESSOR INDEX TABLE, SORTED BY PREDECESSOR,      MCG      32    A
MCG     *        LATE START TIME, AND SUCCESSOR ORDINAL, AND A COUNT OF THE        MCG      33    A
MCG     *        NUMBER OF PREDECESSORS OF EACH INSTRUCTION.                       MCG      34    A
MCG     *        THE PRED COUNT, INDEX TO *SIT*, ETC ARE KEPT IN THE LINK WORD     MCG      35    A
MCG     *        OF AN *IL* INSTRUCTION (R1+3 ) IN "I." FORMAT.                    MCG      36    A
MCG     *        THE ISSUE ( REMOVAL ) OF AN INSTRUCTION FROM THE NETWORK          MCG      37    A
MCG     *        REDUCES THE COUNT OF REMAINING PREDECESSORS ( CONSTRAINTS )       MCG      38    A
MCG     *        ON THE SUCCESSORS OF THE ISSUED INSTRUCTION.                      MCG      39    A
MCG     *        WHEN THE PRED COUNT BECOMES ZERO THE INSTRUCTION IS ADDED TO      MCG      40    A
MCG     *        THE ISSUE CANDIDATE LIST, AND EVENTUALLY ISSUED.                  MCG      41    A
MCG     *        ALSO CALCULATED IN *BDT*, IS THE LATE START TIME, WHICH IS        MCG      42    A
MCG     *        THE LATEST TIME AT WHICH THE INSTRUCTION CAN BE ISSUED WITHOUT    MCG      43    A
MCG     *        INCREASING THE TOTAL TIME ( *SEQTIME* ) IT TAKES TO GET TO        MCG      44    A
MCG     *        THE *EOQ*.                                                        MCG      45    A
MCG     *                                                                          MCG      46    A
MCG     *        TO GENERATE CODE FOR A SEQUENTIAL MACHINE ( I.E. CPU @ 6400 )     MCG      47    A
MCG     *        IT IS SUFFICIENT TO CONSIDER ONLY THE REGISTER ALLOCATION (JAM    MCG      48    A
MCG     *        PROBLEM, AND ATTEMPT TO MINIMIZE THE CODE LENGTH AND NUMBER       MCG      49    A
MCG     *        OF STORES TO COMPILER GENERATED TEMPORARIES.                      MCG      50    A
MCG     *        FOR A MULTI-FUNCTIONAL UNIT MACHINE ( 6600, 7600 ) IT IS          MCG      51    A
MCG     *        NECESSARY TO ALSO CONSIDER TIMING INFORMATION AND SIMULATE        MCG      52    A
MCG     *        THE CPU ( TEMPORAL FUNCTIONAL UNIT AND REGISTER AVAILABILITY )    MCG      53    A
MCG     *                                                                          MCG      54    A
MCG     *        THE FORMER PROBLEM HAS A THEORETICAL SOLUTION ( AHO, ULLMAN -     MCG      55    A
MCG     *        1970 ) WHICH IS NOT COMBINATORIAL IN NATURE WHEN THE NETWORK      MCG      56    A
MCG     *        IS A TREE ( NO MULTIPLE USES OF THE RESULT OF AN INSTRUCTION )    MCG      57    A
MCG     *        , AS OPPOSED TO A DIRECTED ACYCLIC GRAPH.                         MCG      58    A
MCG     *                                                                          MCG      59    A
MCG     *        THE PROBLEM OF PRODUCING AN OPTIMAL SCHEDULE FOR A NETWORK        MCG      60    A
MCG     *        WITH N TASKS AND K PROCESSORS ( K = 9,10 IN OUR CASE )            MCG      61    A
MCG     *        HAS NO GENERAL SOLUTION.  MOST OF THE CURRENT METHODS ARE         MCG      62    A
MCG     *        COMBINATORIAL IN NATURE ( AND IN THE NUMBER OF STEPS ).           MCG      63    A
MCG     *        IN ORDER TO KEEP COMPILATION TIME FINITE AND REASONABLE           MCG      64    A
MCG     *        WE CONTENT OURSELVES WITH A 1 PASS ALGORITHM.                     MCG      65    A
MCG     *        THE BASIC ALGORITHM FOR MAINTAINING THE ISSUE CANIDATE LIST IS SIMILAR MCG  66    A
MCG     *        SIMILAR TO THE "TOPOLOGICAL SORT" DESCRIBED IN VOL I OF KNUTH.    MCG      67    A
MCG     *        THE SELECTION CRITERIA FOR INSTRUCTION IN SCHEDULING MODE         MCG      68    A
MCG     *        IS "EARLIEST START TIME".                                         MCG      69    A
MCG     *                                                                          MCG      70    A
MCG     *        THE FOLLOWING ARE THE TIMING FORMULA OF THE 66/7600 -             MCG      71    A
MCG     *                                                                          MCG      72    A
MCG     *        T(IS) = MAX( T , TRA(RESULT) , TFU(F) )          ISSUE TIME       MCG      73    A
MCG     *        T(EX) = MAX( T(IS) , TRA(OPERANDS) )             EXECUTE TIME     MCG      74    A
MCG     *               T(IS) = T(EX)  FOR A 7600                                  MCG      75    A
MCG     *        TRA(REGNO) = MAX( T(EX)+FT[DI] , TLU(REGNO) )    RESULT AVAIL     MCG      76    A
MCG     *        TFU(F) = TRA(REGNO) + 1                          FUNIT AVAIL      MCG      77    A
MCG     *        TLU(OPERANDS) = MAX( TLU(OPERANDS) , T(EX) )     LAST USE         MCG      78    A
MCG     *        T = T(EX) + N.PARCELS ( + IWTIME )               MIN ISSUE TIM    MCG      79    A
MCG     *                                                                          MCG      80    A
MCG     *        AT PRESENT THE SCHEDULER IGNORES THIRD ORDER CONFLICTS AND THE    MCG      81    A
MCG     *        FORMULAS IT USES ARE -                                            MCG      82    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG      *         TRA(REGNO) = T(EX) + FT[DI]                                  MCG          83     A
MCG      *         TRA(REGNO) = T(EX) + 1   WHEN A REGISTER IS RELEASED.        MCG          84     A
MCG                                                                             MCG          85     A
MCG                LIST   -R                                                    MCG          86     A
MCG                                                                             MCG          87     A
MCG       BOOL     EQU    2           ORDINAL OF BOOLEAN UNIT                   MCG          88     A
MCG       RN.X0    EQU    20B         *SO* REGISTER NUMBER FOR X0               MCG          89     A
MCG       RN.A0    EQU    10B         *SO* REGISTER NUMBER FOR A0               MCG          90     A
MCG       AMX      EQU    -10B        RN.A0-RN.X0  , X TO A CONVERSION          MCG          91     A
MCG       T(INF)   EQU    377777B     *INFINITY*                                MCG          92     A
MCG       MAX.GX   EQU    2           MAX NUMBER OF GENERATED XMITS             MCG          93     A
MCG       DELTAT   EQU    4           MIN DELTA T FOR *LST* CHECK IN *SNI*      MCG          94     A
MCG                                                                             MCG          95          I
          -CCG7
MCG       MX.TXT   ENTRY. 100B        MAX *TXT* LENGTH                          MCG          96          I
          -CCG7
MCG                SPACE  2                                                     MCG          97     A
MCG      *         TYIOM - TYPE I OPERAND MASK AND SHIFT TABLE                  MCG          98     A
MCG                                                                             MCG          99     A
MCG                ENTRY  TYIOM                                                 MCG         100          I
          -CCGNC
MCG                ENTRY  TYIOM#                                                CCGNC       771     A
MCG       TYIOM#   BSS    0                                                     CCGNC       772     A
MCG       TYIOM    BSS    0                                                     MCG         101     A
MCG                VFD    12/2020B,16/-1,16/-1,16/0   RJ, RK                    MCG         102     A
MCG                VFD    12/2020B,16/0,16/-1,16/0    RK                        MCG         103     A
MCG                VFD    12/2000B,16/-1,16/-1,16/-1  RJ, RK, RI                MCG         104     A
MCG                                                                             CCGA073       8     A
MCG                ENTRY  REG=A0                                                CCGA073       9     A
MCG       REG=A0   DATA   8,7         A0 OR B7 FOR FPLI                         CCGA073      10     A
MCG       FXR      TITLE  MACRO DEFINITIONS                                     MCG         105     A
MCG      **        FX. - VALUES OF THE REGISTERS FOR *FXR*                      MCG         106     A
MCG                                                                             MCG         107     A
MCG                ECHO   1,N=(1,2,3,4,5,6,7)                                   MCG         108     A
MCG       FX.X_N   EQU    1S_N*1S8                                              MCG         109     A
MCG       FX.X0    EQU    FX.X7*2                                               MCG         110     A
MCG       FX.XT    EQU    1S8+1S17    SEARCH TERMINATOR AND EXPONENT BIT        MCG         111     A
MCG                                                                             MCG         112     A
MCG                ECHO   1,N=(0,1,2,3,4,5,6,7)  B-REGISTER BIT VALUES          MCG         113     A
MCG       FX.B_N   EQU    1S_N                                                  MCG         114     A
MCG                                                                             MCG         115     A
MCG      *         REGISTER CLASSES ( BIT VALUE MASKS )                         MCG         116     A
MCG                                                                             MCG         117     A
MCG       FB       EQU    2*FX.B7-1         ALL B REGISTERS                     MCG         118     A
MCG       FX       EQU    2*FX.X0-FB-1      ALL X REGISTERS                     MCG         119     A
MCG                                                                             MCG         120     A
MCG       FX.LD    EQU    FX.X1+FX.X2+FX.X3+FX.X4+FX.X5   LOAD REGISTERS        MCG         121     A
MCG       FX.ST    EQU    FX.X6+FX.X7       STORE REGISTERS                     MCG         122     A
MCG       FX.SCR   EQU    FX.X0+FX.LD+FX.ST ALL REGISTERS                       MCG         123     A
MCG                                                                             MCG         124     A
MCG      *         FXRA - REGISTER TRANSLATION TABLE ( SEARCH ORDER )           MCG         125     A
MCG                                                                             MCG         126     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG       FXRA     BSS    0                                                    MCG        127     A
MCG                ECHO   1,N=(0,7,6,5,4,3,2,1)                                MCG        128     A
MCG                VFD    42/FX.X_N,18/RN.X0+N                                 MCG        129     A
MCG                CON    -1          TABLE TERMINATOR                         MCG        130     A
MCG       FXR      SPACE  2                                                    MCG        131     A
MCG      **        FXR - CALL *FXR* TO FIND A RESULT REGISTER IN CLASS *RC*    MCG        132     A
MCG      *         CALL -                                                      MCG        133     A
MCG      *         FXR    RC,SEXIT,FEXIT                                       MCG        134     A
MCG      *                RC = REGISTERS IN CLASS ( LD , ST , SCR )            MCG        135     A
MCG      *                SEXIT = SUCCESS EXIT                                 MCG        136     A
MCG      *                FEXIT = FAIL EXIT ( FALL THROUGH IF ABSENT )         MCG        137     A
MCG                                                                            MCG        138     A
MCG       FXR      MACRO  RC,SEXIT,FEXIT                                       MCG        139     A
MCG                SA1    AXR                                                  MCG        140     A
MCG                IF     -REG,RC,1                                            MCG        141     A
MCG                SX5    FX.RC+FX.XT                                          MCG        142     A
MCG                BX7    X1*X5       AVAIL = AXR & CLASS                      MCG        143     A
MCG                NX6    B3,X7                                                MCG        144     A
MCG                SA3    B3+FXRA+16-47                                        MCG        145     A
MCG                IFGE   .CPU,74                                   ?66/76     MCG        146     A
MCG                SA2    TRA+X3                                               MCG        147     A
MCG                SB2    X2                                                   MCG        148     A
MCG                SB3    X3                                                   MCG        149     A
MCG                LE     B2,B5,SEXIT IF TRA(REGNO) @ T(IS)                    MCG        150     A
MCG                ELSE   2                                         '66/76     MCG        151     A
MCG                SB3    X3                                                   MCG        152     A
MCG                GT     B3,SEXIT                                             MCG        153     A
MCG                RJ     FXR                                                  MCG        154     A
MCG                SB3    X3                                                   MCG        155     A
MCG                IFC    NE,/FEXIT//,2                                        MCG        156     A
MCG                MI     X3,FEXIT                                             MCG        157     A
MCG                ELSE   1                                                    MCG        158     A
MCG                PL     X3,SEXIT                                             MCG        159     A
MCG                ENDM                                                        MCG        160     A
MCG       DEBUG    TITLE  DEBUGGING FACILITIES                                 MCG        161     A
MCG      **        PRNT - PRINT OUT *THING* AT *LAB* IF *THING* OR *LAB* IS    MCG        162     A
MCG      *         MENTIONED ON A *TRACER* DIRECTIVE.                          MCG        163     A
MCG      *         TO DEFINE *PRNT* CALL LABELS -                              MCG        164     A
MCG      *         TRACER (HEAD,II,FINAL,RUI,TIMES,ICL)                        MCG        165     A
MCG                                                                            MCG        166     A
MCG       PRNT     MACRO  LAB,THING,SAVPBI                                     MCG        167     A
MCG                LOCAL  APL        *** DESTROY B2,X6,A6 IF SAVPBI " 0        MCG        168     A
MCG                IF     -DEF,/DEBUG/LAB,1                                    MCG        169     A
MCG       X        IF     DEF,/DEBUG/THING                                     MCG        170     A
MCG                IFC    NE,/SAVPBI//,3                                       MCG        171     A
MCG                SB2    "DI"+B1                                              MCG        172     A
MCG                SX6    B2-"TB"                                              MCG        173     A
MCG                SA6    PBI+1                                                MCG        174     A
MCG      +         RJ     CPR                                                  MCG        175     A
MCG      -         VFD    30/APL                                               MCG        176     A
MCG                USE    DEBUG                                                MCG        177     A
MCG       APL      CON    0,Z.THING,*+2,0,10H LAB                              MCG        178     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                USE    0                                                    MCG        179    A
MCG        X       ENDIF                                                       MCG        180    A
MCG                ENDM                                                        MCG        181    A
MCG                                                                            MCG        182    A
MCG        DBG     IF     DEF,DEBUG                                            MCG        183         I
          -FCC1056
MCG        .T      IFNE   TEST,0                                               FCC1056      7    A
MCG                USE    DEBUG                                                MCG        184    A
MCG        Z.ICL   CON    1          ISSUE CANIDATE LIST                       MCG        185    A
MCG        Z.II    CON    2          ISSUED INSTRUCTION                        MCG        186    A
MCG        Z.RUI   CON    3          REGISTER USEAGE INFO                      MCG        187    A
MCG        Z.TIMES CON    4          FU AND X-REG AVAIL TIMES                  MCG        188    A
MCG        Z.FINAL CON    5          FINAL SUMMARY                             MCG        189    A
MCG        Z.HEAD  CON    6          LISTING HEADER ( II AND FINAL OPTIONS ONLY MCG       190    A
MCG                                                                            MCG        191    A
MCG       **       CPR - CALL PRINT ROUTINE                                    MCG        192    A
MCG                                                                            MCG        193    A
MCG        CPR0    RJ     =XRESET=                                             MCG        194         I
          -CCG7
MCG        CPR0    RJ     =XRSR=                                               CCG7       125    A
MCG        CPR     ROUTINE                                                     MCG        195    A
MCG                RJ     =XSAVE=                                              MCG        196         I
          -CCG7
MCG                RJ     =XSVR=                                               CCG7       126    A
MCG                SA1    CPR                                                  MCG        197    A
MCG                SA4    O.TXT                                                MCG        198    A
MCG                LX1    30                                                   MCG        199    A
MCG                SA2    X1-1                                                 MCG        200    A
MCG                BX6    X4                                                   MCG        201    A
MCG                SA6    X2         [APL] = [O.TXT]                           MCG        202    A
MCG                SA1    A6                                                   MCG        203    A
MCG       +        RJ     =XPRNTMCG                                            MCG        204    A
MCG                SA1    EOQA                                                 MCG        205    A
MCG                PL     X1,CPR0                                              MCG        206    A
MCG                BX6    -X1                                                  MCG        207    A
MCG                SA6    A1                                                   MCG        208    A
MCG                RJ     =XRESET=                                             MCG        209         I
          -CCG7
MCG                RJ     =XRSR=                                               CCG7       127    A
MCG                SNAPT  TXT,CPR-ERR                                          MCG        210    A
MCG                EQ     CPR                                                  MCG        211    A
MCG                                                                            MCG        212    A
MCG                USE    0                                                    MCG        213    A
MCG        DBG     ENDIF                                                       MCG        214         I
          -FCC1056
MCG        .T      ENDIF                                                       FCC1056      8    A
MCG        FLAGS   TITLE  VARIABLES                                           MCG        215    A
MCG                USE    /MCG/                                                MCG        216    A
MCG        N.INST  BSS    1          NUMBER OF INSTRUCTIONS ( L.TXT/4 )       MCG        217    A
MCG        N.SP    BSS    1          -( NUMBER OF STORE PREDECESSOR OPS )     MCG        218    A
MCG        SEQTIME BSS    1          CRITICAL PATH TIME                       MCG        219    A
MCG                                                                            MCG        220    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG       ICL      BSS    1            ISSUE CANDIDATE LIST POINTER                      MCG        221     A
MCG       RMR      BSSZ   1            24/0,18/CA,18/A-REG NUMBER                        MCG        222     A
MCG                VFD    12/1776B,12/0,18/T(INF),18/T(INF)                              MCG        223     A
MCG       PBI      BSS    2            PREVIOUS BEST INST INFO ( PB. FORMAT )            MCG        224     A
MCG                                                                                      MCG        225     A
MCG       DT       CON    4            DELTA T FOR *LST* CHECK                           MCG        226     A
MCG       IWT      CON    .IWT         INTERWORD TIME ( =0 IF NOT A SLOW 6600 )          MCG        227     A
MCG                                                                                      MCG        228     A
MCG       AXR      CON    1S58+FX+FB   AVAILABLE X AND B REGISTERS                       MCG        229     A
MCG       ALR      CON    FX.LD+FX.XT  AVAILABLE LOAD REGISTERS                          MCG        230     A
MCG       STL      CON    -FX.X6       STORE LOCKOUT FLAG                                MCG        231     A
MCG                                                                                      MCG        232     A
MCG       RVT      BSSZ   24           REGISTER VALUE TABLE ( R-NUMBER IN REGISTER       MCG        233     A
MCG      *                             1/LOCK,59/INDEX TO *TXT* OF *RI* IN REG           MCG        234     A
MCG                                                                                      MCG        235     A
MCG                CON    1S59+T(INF)        TRA-1  ( FOR *FXR* )                        MCG        236     A
MCG       F.CLC    BSS    0           ?FWA OF CLEARED CELLS                             MCG        237     A
MCG       TRA      BSS    24           TIME REGNO(I) AVAILABLE ( BAX ORDER )             MCG        238     A
MCG       ARA      BSS    8            A-REGISTER ADDRESS INFO ( [IH WORD] )             MCG        239     A
MCG       TFU      BSS    12           TIME FU(I) AVAILABLE                              MCG        240     A
MCG       LST      BSS    2            COMPLETION TIME OF LAST STORE ( 6600 )            MCG        241     A
MCG       LLT      EQU    LST+1        COMPLETION TIME OF LAST LOAD                      MCG        242     A
MCG       LXR      BSS    1            LOCKED X AND B-REGISTERS                          MCG        243     A
MCG       NGX      BSS    1            N.GENERATED XMITS                                 MCG        244     A
MCG       T        BSS    1            CURRENT SIMULATION TIME ( MINOR CYCLES )          MCG        245     A
MCG       PC       ENTRY. 0            PARCEL COUNT ( 0 - 3 )                            MCG        246         I
         -CCGNC
MCG       PC       ENTRY. 0,#          PARCEL COUNT ( 0 - 3 )                            CCGNC      773     A
MCG       TRS      BSS    1            BIT I SET IF A RESULT WILL BE READY AT T+I        MCG        247     A
MCG       L.CLC    EQU    *-F.CLC     '                                                 MCG        248     A
MCG                                                                                      MCG        249     A
MCG       JLR      CON    0            BIT(47-REGNO) SET FOR REGISTERS LOCKED            MCG        250     A
MCG      *                             UNTIL NEXT UNCONDITIONAL JUMP.                    MCG        251     A
MCG       LRM      BSS    2            LOCKED REG MODE, INITIAL PIT LENGTH               MCG        252     A
MCG       EOQA     CON    0            ADDRESS OF *EOQ* PROCESSOR                        MCG        253     A
MCG       IIP      BSS    2            ISSUED INST PTR ( *PIT* ADDR ) / O.PIT            MCG        254     A
MCG       EIW      BSS    1            -(NUMBER OF EXTRA PIT WORDS LEFT) JAM MODE        MCG        255     A
MCG                                                                                      MCG        256     A
MCG      *         RBV - REGISTER BIT VALUE TABLE ( *SO* REGNO ORDER )                   MCG        257     A
MCG                                                                                      MCG        258     A
MCG       RBV      BSS    0                                                              MCG        259     A
MCG                ECHO   1,N=(0,1,2,3,4,5,6,7)                                          MCG        260     A
MCG                CON    FX.B_N                                                         MCG        261     A
MCG                BSSZ   8            A-REGISTERS                                       MCG        262     A
MCG                ECHO   1,N=(0,1,2,3,4,5,6,7)                                          MCG        263     A
MCG                CON    FX.X_N                                                         MCG        264     A
MCG                                                                                      MCG        265     A
MCG                USE    /TABLES/                                                       MCG        266         I
         -CCG1
MCG                USE    /CCG.SCR/                                                      CCG1       284         I
         -CCG18
MCG                USE    /CCGSCR/                                                       CCG18      173     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG       SVL      BSS    100B        SAVED LINK WORDS                                MCG        267     A
MCG                USE    0                                                           MCG        268     A
MCG                                                                                   MCG        269     A
MCG       DTT      BSS    1           DELTA T TABLE                                   MCG        270     A
MCG                ECHO   1,D=(DELTAT,8,16,32)                                        MCG        271     A
MCG                VFD    15/D                                                        MCG        272     A
MCG       TABLES   TITLE  TABLE DEFINATIONS                                           MCG        273     A
MCG      **        USES INDEX ( SUCCESSOR ) TABLE                                     MCG        274     A
MCG                                                                                   MCG        275     A
MCG                DESCRIBE U.,60                                                     MCG        276     A
MCG       LAST     DEFINE 1           "0 IF LAST EDGE OF A GROUP                      MCG        277     A
MCG       EQV      DEFINE 1           EQV  ( DATA INTERFERENCE ) LINK                 MCG        278           I
         -FCC0448
MCG                DEFINE 4                                                           MCG        279           I
         -FCC0448
MCG       RDL      DEFINE 1           REGISTER REDEFINITION LINK                      FCC0448    147     A
MCG       EQV      DEFINE 1           DATA INTERFERENCE LINK                          FCC0448    148     A
MCG                DEFINE 3                                                           FCC0448    149     A
MCG       PRED     DEFINE 18          INDEX TO *TXT* OF PRED                          MCG        280     A
MCG       LST      DEFINE 18          LATE START TIME OF SUCCESSOR                    MCG        281     A
MCG       SUCC     DEFINE 18          INDEX TO *TXT* OF SUCCESSOR                     MCG        282     A
MCG       INFO     SPACE  2                                                           MCG        283     A
MCG      *         LINK INFORMATION WORD    ( INFO SETUP BY *BDT* )                   MCG        284     A
MCG                                                                                   MCG        285     A
MCG                DESCRIBE I.,60                                                     MCG        286     A
MCG       II       DEFINE 1           "0 IF INSTRUCTION IS ISSUED ( POSTED )          MCG        287     A
MCG       IC       DEFINE 1           "0 IS INSTRUCTION ON ISSUE CANDIDATE LIST       MCG        288     A
MCG                DEFINE 1                                                           MCG        289     A
MCG       USES     DEFINE 9                                                           MCG        290     A
MCG       INDX     DEFINE 15          INDEX TO *TREE* OF SUCCESSORS OF INSTRUCTION    MCG        291     A
MCG       LST      DEFINE 15          LATE START TIME                                 MCG        292     A
MCG       NPRED    DEFINE 18          NUMBER OF UNISSUED PREDECESSORS                 MCG        293     A
MCG                                                                                   MCG        294     A
MCG       LINK     DEQU   NPRED       INDEX TO NEXT ON ISSUE CANIDATE LIST            MCG        295     A
MCG       REG      DEQU   NPRED       REGISTER NUMBER THAT *RI* IS IN ( ISSUED )      MCG        296     A
MCG       PBI      SPACE  2                                                           MCG        297     A
MCG      **        FORMAT OF SAVED PREVIOUS BEST INSTRUCTION INFO                     MCG        298     A
MCG                                                                                   MCG        299     A
MCG                DESCRIBE PB.,60    FIRST WORD                                      MCG        300     A
MCG       REG      DEFINE 12          PACKED REGISTER ORDINAL                         MCG        301     A
MCG                DEFINE 12                                                          MCG        302     A
MCG       TEX      DEFINE 18          EXECUTION TIME                                  MCG        303     A
MCG       TIS      DEFINE 18          ISSUE TIME                                      MCG        304     A
MCG                                                                                   MCG        305     A
MCG                DESCRIBE PB.,60    SECOND WORD                                     MCG        306     A
MCG       FU       DEFINE 12          PACKED FUNIT ORDINAL                            MCG        307     A
MCG                DEFINE 12                                                          MCG        308     A
MCG       PI       DEFINE 18          INDEX TO *TXT* OF PREVIOUS INST                 MCG        309     A
MCG       LI       DEFINE 18          INDEX TO *TXT* OF PREVIOUS BEST INST            MCG        310     A
MCG       REGAS    SPACE  3                                                           MCG        311     A
MCG      **        THE FOLLOWING REGISTER ASSIGNMENTS ARE HELD OVER MOST OF *MCG*     MCG        312     A
MCG      *                                                                           MCG        313     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG      *         (B1) = 1  , B2 - SCRATCH                                    MCG        314     A
MCG      *         B3 - REGNO                                                  MCG        315     A
MCG      *         B4 - T(EX) EXECUTION START TIME OF CURRENT INSTRUCTION      MCG        316     A
MCG      *         B5 - T(IS) ISSUE TIME OF CURRENT INSTRUCTION                MCG        317     A
MCG      *         B6 - TIS(PBI) IN *SNI*, *DRR*, ETC., ELSE SCRATCH           MCG        318     A
MCG       TB       MICRO  1,,/B7/     O.TXT + 3                                MCG        319     A
MCG       PI       MICRO  1,,/A0/     *TXT* INDEX OF PREVIOUS INSTRUCTION      MCG        320     A
MCG       DI       MICRO  1,,/A4/     ADDRESS OF DESCRIPTOR OF CURRENT INST    MCG        321     A
MCG       RN       MICRO  1,,/-X0*/   R-NUMBER EXTRACTOR MASK                 MCG        322     A
MCG                                                                           MCG        323     A
MCG      *         THE OTHER X-REGISTERS MAY BE USED AS SCRATCH .              MCG        324     A
MCG       MCG      TITLE  CONTROL / INITIALIZATION                             MCG        325     A
MCG                LIST   F,R                                                  MCG        326     A
MCG                                                                           MCG        327     A
MCG      **        MCG - MACHINE CODE GENERATOR ( CONTROL / MAIN LOOP )        MCG        328     A
MCG      *                                                                    MCG        329     A
MCG      *         ENTRY  (X6) = LRM , " 0 IF LOCKED X-REGISTERS               MCG        330     A
MCG      *                                                                    MCG        331     A
MCG      *         EXIT   (X6) = FAIL , " 0 IF SCHEDULER COULDNT CODE THE      MCG        332     A
MCG      *                SEQUENCE WITH LOCKED REGISTERS.                      MCG        333     A
MCG                                                                           MCG        334     A
MCG       MCG      ENTRY. **                                                   MCG        335           I
          -CCGNC
MCG       MCG      ENTRY. **,#                                                 CCGNC      774     A
MCG                SA1    L.PIT                                                MCG        336     A
MCG                SX7    X1+                                                  MCG        337     A
MCG                SA6    LRM        LRM = FLAG                                MCG        338     A
MCG                SA7    A6+1       IPL = L.PIT   */ SAVE INITIAL *PIT* LENGTH MCG        339     A
MCG                                                                           MCG        340     A
MCG       #MD      IFGE   .CPU,74                                   ?66/76    MCG        341     A
MCG                SA5    L.TXT                                                MCG        342     A
MCG                SA1    =XOPT2                                               MCG        343           I
          -CCG7
MCG                SA1    =XCO.OPT                                             CCG7       128           I
          -CCGNC
MCG                SA1    =XHO$OPT                                             CCGNC      775     A
MCG                SB2    X5-4*64                                              MCG        344     A
MCG                SA2    =XWB                                                 MCG        345           I
          -CCG7
MCG                SA2    =XCG.OPTL                                            CCG7       129           I
          -CCGNC
MCG                SA2    =XCC$OPTL                                            CCGNC      776     A
MCG                PL     B2,MCG9    IF L.TXT \ 4*64                           MCG        346     A
MCG                                                                           MCG        347     A
MCG                NZ     X1,MCG1    IF OPT2 " 0                               MCG        348           I
          -CCG7
MCG                SX3    X2-1                                                 MCG        349           I
          -CCG7
MCG                NZ     X3,MCG9    IF WB " 1     */ NOT OPT=1 & INNERMOST LOOP MCG      350           I
          -CCG7
MCG                MI     X1,MCG1    IF OPT = 2                                CCG7       130     A
MCG                ZR     X2,MCG9    IF CG.OPTL = 0  */ NOT AN OPTIMIZABLE LOOP CCG7      131           I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
-CCGNC
MCG                ZR     X2,MCG9      IF CG$OPTL = 0  */ NOT AN OPTIMIZABLE LOOP          CCGNC        777     A
MCG                                                                                         MCG          351     A
MCG        MCG1    SA5    O.TXT                                                             MCG          352     A
MCG                CALL   CRW          COMPUTE REG WIDTH                                    MCG          353          I
-CCGNC
MCG                CALL   CRW#         COMPUTE REG WIDTH                                    CCGNC        778     A
MCG                PRINT  MCG,(*  LINENR,REGW =*2I5),(=XLINENR,X6)                          MCG          354          I
-CCG1
MCG                PRINT  MCG,(*  LINENR ,REGW =*2I5),(=XCO.CSN ,X6)                        CCG1         285          I
-CCGNC
MCG                PRINT  MCG,(*  LINENR ,REGW =*2I5),(=XHO$CSN ,X6)                        CCGNC        779     A
MCG                SB6    X6                                                                MCG          355     A
MCG                SB7    9                                                                 MCG          356     A
MCG                LT     B6,B7,MCG2   IF MAXW @ 8                                         MCG          357     A
MCG                GT     B6,B7,MCG9   IF MAXW > 9                                         MCG          358     A
MCG                                                                                         MCG          359     A
MCG        MCG2    SA1    DTT+1                                                             MCG          360     A
MCG                SB7    B7-B1                                                             MCG          361     A
MCG                BX7    X1           DTT = DTT(2)  */ SET INCREMENT VALUES                MCG          362     A
MCG                LT     B6,B7,MCG3   IF MAXW < 8                                         MCG          363     A
MCG                AX7    15           DTT = SHIFT(DTT,-15)  */ SET FOR 3 TRIES             MCG          364     A
MCG        MCG3    SA7    A1-B1                                                             MCG          365     A
MCG                MX6    0                                                                 MCG          366     A
MCG                CALL   BDT          FORM THE DEPENDENCY GRAPH                            MCG          367          I
-CCGNC
MCG                CALL   BDT#         FORM THE DEPENDENCY GRAPH                            CCGNC        780     A
MCG                                                                                         MCG          368     A
MCG        *       SAVE THE LINK WORDS                                                      MCG          369     A
MCG                                                                                         MCG          370     A
MCG                SA1    O.TXT                                                             MCG          371     A
MCG                SA2    N.INST                                                            MCG          372     A
MCG                SA5    X1+3                                                              MCG          373     A
MCG                SB2    X2-1                                                              MCG          374     A
MCG                BX6    X5                                                                MCG          375     A
MCG                SA6    SVL                                                               MCG          376     A
MCG        MCG5    SA5    A5+4         FOR I = 1 TO N.INST;  SVL(I) = TXT(4*I)              MCG          377     A
MCG                SB2    B2-B1                                                             MCG          378     A
MCG                BX6    X5                                                                MCG          379     A
MCG                SA6    A6+B1                                                             MCG          380     A
MCG                NZ     B2,MCG5                                                           MCG          381     A
MCG                                                                                         MCG          382     A
MCG        *       TRY SCHEDULING THE SEQUENCE WITH DIFFERENT LOOK AHEAD VALUES             MCG          383     A
MCG        *       UNTIL IT CAN BE CODED SUCCESSFULLY ( *SIS* EXITS TO *MCG14* )            MCG          384     A
MCG                                                                                         MCG          385     A
MCG        MCG6    SA1    DTT                                                               MCG          386     A
MCG                MX3    -15                                                               MCG          387     A
MCG                BX6    -X3*X1       DT = DTT & 77777B  */ SET LOOK AHEAD VALUE           MCG          388     A
MCG                SA6    DT                                                                MCG          389     A
MCG                SX7    MCG14                                                             MCG          390     A
MCG                RJ     SIS          SCHEDULE THE SEQUENCE                                MCG          391     A
MCG                                                                                         MCG          392     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG       *          FAILURE, RESTORE LINK WORDS AND TRY WITH A SMALLER DELTA T.          MCG        393    A
MCG                                                                                       MCG        394    A
MCG                  SA1    DTT                                                           MCG        395    A
MCG                  AX1    15                                                            MCG        396    A
MCG                  BX6    X1          DTT = SHIFT(DTT,-15)  */ ADVANCE TO NEXT IN       MCG        397    A
MCG                  SA6    A1                                                            MCG        398    A
MCG                  ZR     X1,MCG11    IF DTT = 0    */ END OF DELTA T TABLE             MCG        399    A
MCG                                                                                       MCG        400    A
MCG                  SA5    SVL                                                           MCG        401    A
MCG                  SA1    O.TXT                                                         MCG        402    A
MCG                  SA2    N.INST                                                        MCG        403    A
MCG                  BX6    X5                                                            MCG        404    A
MCG                  SB2    X2-1                                                          MCG        405    A
MCG                  SA6    X1+3                                                          MCG        406    A
MCG                  SB4    4                                                             MCG        407    A
MCG        MCG7      SA5    A5+B1       FOR I = 1 TO N.INST;  TXT(4*I) = SVL(I)           MCG        408    A
MCG                  SB2    B2-B1                                                         MCG        409    A
MCG                  BX6    X5                                                            MCG        410    A
MCG                  SA6    A6+B4                                                         MCG        411    A
MCG                  NZ     B2,MCG7                                                       MCG        412    A
MCG                                                                                       MCG        413    A
MCG                  EQ     MCG6                                                          MCG        414    A
MCG                                                                                       MCG        415    A
MCG        MCG9      SX6    DELTAT                                                        MCG        416    A
MCG                  SA6    DT          DT = DELTAT   */ SET DEFAULT DELTA T              MCG        417    A
MCG                                                                                       MCG        418    A
MCG        #MD       ENDIF                                                ?66/76          MCG        419    A
MCG                                                                                       MCG        420    A
MCG       *          INITIALIZE FOR SCHEDULING MODE                                       MCG        421    A
MCG                                                                                       MCG        422    A
MCG        MCG10     MX6    0                                                             MCG        423    A
MCG                  CALL   BDT         FORM THE DEPENDENCY GRAPH                         MCG        424         I
          -CCGNC
MCG                  CALL   BDT#        FORM THE DEPENDENCY GRAPH                         CCGNC      781    A
MCG                  SX7    MCG14                                                         MCG        425    A
MCG                  RJ     SIS         TRY AND SCHEDULE THE SEQUENCE                     MCG        426    A
MCG                                                                                       MCG        427    A
MCG       *          SCHEDULER FAILED, TRY *JAM* IF NOT *LOCKED X MODE*.                  MCG        428    A
MCG                                                                                       MCG        429    A
MCG        MCG11     SA1    LRM                                                           MCG        430    A
MCG                  SX6    B1          FAIL = 1                                          MCG        431    A
MCG                  MX7    0                                                             MCG        432    A
MCG                  SA7    L.TREE      L.TREE = 0                                        MCG        433    A
MCG                  NZ     X1,MCG      IF LRM " 0    */ LOCKED X-MODE                    MCG        434    A
MCG                  SA1    ALR                                                           FCC0603     94    A
MCG                  SX2    FX.LD                                                         FCC0603     95    A
MCG                  BX3    X2*X1                                                         FCC0603     96    A
MCG                  IX4    X2-X3                                                         FCC0603     97    A
MCG                  NZ     X4,MCG      IF ALR&FX.LD " FX.LD  */ LOCKED A-REGISTERS       FCC0603     98    A
MCG                                                                                       MCG        435    A
MCG                  RJ     /JAM/JAM    JAM CODE THE SEQUENCE                             MCG        436    A
MCG                                                                                       MCG        437    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG       *         ACCUMULATE *FL* USED                                    MCG          438          I
          -RSM2763X
MCG                                                                         MCG          439          I
          -RSM2763X
MCG        MCG14    SA1    L.TXT                                            MCG          440          I
          -RSM2763X
MCG                 SA3    =XOPT2                                           MCG          441          I
          -RSM2763X
MCG                 SA2    =XMX.TXT                                         MCG          442          I
          -RSM2763X
MCG                 ZR     X3,MCG16    IF OPT2 = 0                          MCG          443          I
          -RSM2763X
MCG                 IX6    X1-X2                                            MCG          444          I
          -RSM2763X
MCG                 MI     X6,MCG16    IF L.TXT < MX.TXT                    MCG          445          I
          -RSM2763X
MCG                 BX7    X1                                               MCG          446          I
          -RSM2763X
MCG                 SA7    A2          MX.TXT = L.TXT                       MCG          447          I
          -RSM2763X
MCG                 CALL   AMU=        ACCUMULATE MEMORY USED               MCG          448          I
          -RSM2763X
MCG                                                                         MCG          449          I
          -RSM2763X
MCG        MCG16    BSS    0                                                MCG          450          I
          -RSM2763X
MCG        MCG14    BSS    0                                                RSM2763X     368     A
MCG                 PRNT   JAM,FINAL                                        MCG          451     A
MCG                 SA1    LXR                                              MCG          452     A
MCG                 SX6    0           FAIL = 0                             MCG          453     A
MCG                 SA6    L.TREE      L.TREE = 0                           MCG          454     A
MCG                 ZR     X1,MCG      IF LXR = 0    */ RVT IS ZERO         MCG          455     A
MCG                 RJ     CLR         CLEAR *RVT*                          MCG          456     A
MCG                 EQ     MCG                                              MCG          457     A
MCG         SIS     SPACE  3,14        SIS                                  MCG          458     A
MCG       **        SIS - SCHEDULE INSTRUCTION SEQUENCE                     MCG          459     A
MCG       *                                                                 MCG          460     A
MCG       *         ENTRY  (X7) = SUCCESS EXIT ADDRESS                      MCG          461     A
MCG       *                                                                 MCG          462     A
MCG       *         EXIT   RETURNS TO CALLER IF CANNOT CODE SEQUENCE, ELSE TO MCG        463     A
MCG       *                SUCCESS EXIT.                                    MCG          464     A
MCG                                                                         MCG          465     A
MCG         SIS0    SA1    LRM+1                                            MCG          466     A
MCG                 SX7    X1+         L.PIT = IPL   */ RESTORE *PIT* LENGTH MCG          467     A
MCG                 SA7    L.PIT                                            MCG          468     A
MCG                 RJ     CLR         CLEAR *RVT*                          MCG          469     A
MCG                                                                         MCG          470     A
MCG         SIS     ROUTINE                                                 MCG          471     A
MCG                 SX0    0                                                MCG          472     A
MCG                 RJ     PRS         PRESET STORAGE                       MCG          473     A
MCG        #MD      IFGE   .CPU,74                                 ?66/76   MCG          474     A
MCG                 SA1    DT                                               MCG          475     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SA2    SNIB                                                    MCG        476    A
MCG                LX1    30                                                      MCG        477    A
MCG                BX7    X1+X2       PLUG *DT* IN *SNI*                          MCG        478    A
MCG                SA7    SNI.P                                                   MCG        479    A
MCG        #MD     ENDIF                                          ?66/76           MCG        480    A
MCG                EQ     SIS2                                                    MCG        481    A
MCG                                                                               MCG        482    A
MCG        SIS1    RJ     AIL         ADJUST THE ISSUE LIST                       MCG        483    A
MCG                RJ     DUC         DECREMENT USES COUNTS                       MCG        484    A
MCG                SA4    "DI"                                                    MCG        485    A
MCG                RJ     ASC         ADVANCE THE CLOCKS                          MCG        486    A
MCG                SA5    "DI"-2      R1 = DI - 2                                 MCG        487    A
MCG                RJ     SII         SAVE THE ISSUED INSTRUCTION                 MCG        488    A
MCG                                                                               MCG        489    A
MCG        SIS2    RJ     SNI         SELECT THE NEXT INSTRUCTION TO ISSUE        MCG        490    A
MCG                PL     B3,SIS1     IF REGNO > 0  */ ISSUABLE INSTRUCTION FOUND MCG        491    A
MCG        SIS     SPACE  1,10                                                    MCG        492    A
MCG       *        CHECK FOR A STORE REGISTER JAM, AND GENERATE AN *XMT*          MCG        493    A
MCG       *        IF POSSIBLE.                                                   MCG        494    A
MCG                                                                               MCG        495    A
MCG                SA1    AXR                                                     MCG        496    A
MCG                SA5    NGX                                                     MCG        497    A
MCG                SX2    FX.ST                                                   MCG        498    A
MCG                SX3    FX.SCR                                                  MCG        499    A
MCG                BX2    X2*X1                                                   MCG        500    A
MCG                BX3    X3*X1                                                   MCG        501    A
MCG                NZ     X2,SIS0     IF AXR & FX.ST " 0  */ NOT ST-REG JAM       MCG        502    A
MCG                ZR     X3,SIS0     IF AXR & FX.SCR = 0  */ ALL REGS IN USE     MCG        503    A
MCG                SB3    X5-MAX.GX                                               MCG        504    A
MCG                ZR     B3,SIS0     IF NGX = MAX.GX  */ GEN XMT LIMIT REACHED   MCG        505    A
MCG                                                                               MCG        506    A
MCG       *        FIND A ST-REG WHOSE RESULT CAN BE MOVED TO ANOTHER REGISTER.   MCG        507    A
MCG                                                                               MCG        508    A
MCG                SB4    RN.X0+6     REGF = X6     */ TRY X6                     MCG        509    A
MCG                RJ     CSR                                                     MCG        510    A
MCG                ZR     X7,SIS3     IF CSR(REGF) = 0  */ RESULT CAN BE MOVED    MCG        511    A
MCG                SB4    B4+1        REGF = X7                                   MCG        512    A
MCG                RJ     CSR                                                     MCG        513    A
MCG                NZ     X7,SIS0     IF CSR(REGF) " 0                            MCG        514    A
MCG                                                                               MCG        515    A
MCG        SIS3    SA1    AXR                                                     MCG        516    A
MCG                SA2    T                                                       MCG        517    A
MCG                SX5    FX.SCR+FX.XT                                            MCG        518    A
MCG                BX7    X1*X5       AVAIL = AXR & FX.SCR                        MCG        519    A
MCG                NX6    B3,X7                                                   MCG        520    A
MCG                SB5    X2+         T(IS) = T                                   MCG        521    A
MCG                SA3    B3+FXRA+16-47                                           MCG        522    A
MCG                SA2    TRA+X3                                                  MCG        523    A
MCG                SB2    X2                                                      MCG        524    A
MCG                SB3    X3          REGR = FXR( SCR )  */ RESULT REG            MCG        525    A
MCG                LE     B2,B5,SIS4  IF TRA(REGR) @ T(IS)                        MCG        526    A
MCG                SB5    B2          T(IS) = TRA(REGR)                           MCG        527    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG       SIS4     SB2    OC.XMT                                                  MCG          528     A
MCG                RJ     /JAM/OXI    OUTPUT AN *XMT* INSTRUCTION                 MCG          529     A
MCG                SA2    NGX                                                     MCG          530     A
MCG                SX6    X2+B1       NGX = NGX + 1                               MCG          531     A
MCG                SA6    A2                                                      MCG          532     A
MCG                EQ     SIS2                                                    MCG          533     A
MCG       CSR      SPACE  3,14        CSR                                         MCG          534     A
MCG      **        CSR - CHECK SPECIFIED REGISTER TO SEE IF THE RESULT IN IT      MCG          535     A
MCG      *         CAN BE MOVED TO ANOTHER REGISTER ( ^PRS , ETC).                MCG          536     A
MCG      *                                                                       MCG          537     A
MCG      *         ENTRY  (B4) = REGF , REGNO OF REGISTER TO BE CHECKED.          MCG          538     A
MCG      *                                                                       MCG          539     A
MCG      *         EXIT   (X7) = PSF = 0 IF RESULT IN REGF CAN BE MOVED           MCG          540     A
MCG                                                                              MCG          541     A
MCG       CSR4     LX4    D.PSP-D.PRSP                                            MCG          542     A
MCG                MX6    1                                                       MCG          543     A
MCG                BX7    X6*X4       PSF = PRS[DI] */ MOVABLE IF ^PRS            MCG          544     A
MCG                                                                              MCG          545     A
MCG       CSR      ROUTINE                                                        MCG          546     A
MCG                SA1    RVT+B4      R = RVT(REGF)                               MCG          547     A
MCG                SB6    "TB"-B1                                                 MCG          548     A
MCG                SA4    B6+X1       DI = BASE-1 + R                             MCG          549     A
MCG                LX4    59-D.PIP                                                MCG          550     A
MCG                MI     X4,CSR3     IF PI[DI]     */ PSEUDO INSTRUCTION         MCG          551     A
MCG                LX4    D.PIP-D.PSP                                             MCG          552     A
MCG                PL     X4,CSR4     IF ^PS[DI]                                  MCG          553     A
MCG                                                                              MCG          554     A
MCG      *         RESULT *PS*, CHECK UNISSUED SUCCESSORS OF *R*                  MCG          555     A
MCG                                                                              MCG          556     A
MCG       CSR0     SA5    A4+B1       LI = DI + 1                                 MCG          557     A
MCG                MX7    -I.INDXL    PSF = 1                                     MCG          558     A
MCG                SA2    O.TREE                                                  MCG          559     A
MCG                LX5    -I.INDXP                                                MCG          560     A
MCG                BX6    -X7*X5                                                  MCG          561     A
MCG                SB2    X2                                                      MCG          562     A
MCG                SA3    B2+X6       UI = O.TREE + INDX[LI]                      MCG          563     A
MCG                                                                              MCG          564     A
MCG       CSR1     LX3    -U.SUCCP                                                MCG          565     A
MCG                SA5    "TB"+X3     SI = BASE + SUCC[UI]                        MCG          566     A
MCG                LX5    59-I.IIP                                                MCG          567     A
MCG                MI     X5,CSR2     IF II[SI]     */ ISSUED                     MCG          568     A
MCG                SA1    A5-B1       D = SI - 1                                  MCG          569          I
          -FCC0393
MCG                LX1    59-D.STP                                                MCG          570          I
          -FCC0393
MCG       #DAL     IFEQ   .DAL,0                                                  MCG          571          I
          -FCC0393
MCG                MI     X1,CSR      IF ST[D]      */ UNISSUED *ST*              MCG          572          I
          -FCC0393
MCG       #DAL     ELSE                                                           MCG          573          I
          -FCC0393
MCG                PL     X1,CSR2     IF ^ST[D]                                   MCG          574          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

          -FCC0393
MCG                 LX1    D.STP-D.L2P                                          MCG          575          I
          -FCC0393
MCG                 PL     X1,CSR      IF ^L2[D]                                MCG          576          I
          -FCC0393
MCG        #DAL     ENDIF                                                       MCG          577          I
          -FCC0393
MCG                 SA2    A5-3        R1S = SI - 3                             FCC0393        6     A
MCG                 BX5    X2-X1                                                FCC0393        7     A
MCG                 BX6    "RN"X5                                               FCC0393        8     A
MCG                 NZ     X6,CSR2     IF RI[R1S] " R  */ NOT USED IN *RI*      FCC0393        9     A
MCG                 SA2    A5-B1       D = SI - 1                               FCC0393       10     A
MCG                 LX2    59-D.STP                                             FCC0393       11     A
MCG                 PL     X2,CSR2     IF ^ST[D]                                FCC0393       12     A
MCG                 LX2    D.STP-D.L2P                                          FCC0393       13     A
MCG                 PL     X2,CSR      IF ^L2[D]     */ UNISSUED STORE          FCC0393       14     A
MCG       CSR2      BX6    X3          UL = UI                                  MCG          578     A
MCG                 SA3    A3+B1       UI = UI + 1                              MCG          579     A
MCG                 LX6    U.SUCCP+59-U.LASTP                                   MCG          580     A
MCG                 PL     X6,CSR1     IF ^LAST[UL]                             MCG          581     A
MCG                                                                             MCG          582     A
MCG                 MX7    0           PSF = 0                                  MCG          583     A
MCG                 EQ     CSR                                                  MCG          584     A
MCG                                                                             MCG          585     A
MCG       CSR3      SA5    A4-2        R1 = DI - 2   */ INSTRUCTION IS A *DEF*  MCG          586     A
MCG                 MX6    1                                                    MCG          587     A
MCG                 LX5    58-R1.SOP-SO.LKP                                     MCG          588     A
MCG                 BX7    X6*X5       PSF = SOLK[R1]/2  */ =1 IF A FULL LOCK   MCG          589     A
MCG                 NZ     X7,CSR      IF PSF " 0                               MCG          590     A
MCG                 EQ     CSR0                                                 MCG          591     A
MCG        CLR      SPACE  3,10        CLR                                      MCG          592     A
MCG       **        CLR - CLEAR *RVT*                                           MCG          593     A
MCG                                                                             MCG          594     A
MCG        CLR1     SA7    A6+B1       FOR I = 1 TO 23; RVT(I) = 0              MCG          595     A
MCG                 LX5    1                                                    MCG          596     A
MCG                 SA6    A7+1                                                 MCG          597     A
MCG                 MI     X5,CLR1                                              MCG          598     A
MCG        CLR      ROUTINE                                                     MCG          599     A
MCG                 SX6    B0                                                   MCG          600     A
MCG                 MX5    22/2                                                 MCG          601     A
MCG                 SX7    0                                                    MCG          602     A
MCG                 SA6    RVT+1                                                MCG          603     A
MCG                 EQ     CLR1                                                 MCG          604     A
MCG        PRS      SPACE  3,24        PRS                                      MCG          605     A
MCG       **        PRS - PRESET STORAGE ( INITIALIZATION )                     MCG          606     A
MCG       *                                                                     MCG          607     A
MCG       *         ENTRY  (X7) = *EOQ* PROCESSOR ADDRESS                       MCG          608     A
MCG       *                (X0) = CODE MODE, 0/1 = NORMAL/JAM                   MCG          609     A
MCG                                                                             MCG          610     A
MCG        PRS      ROUTINE                                                     MCG          611     A
MCG                 SA7    EOQA        EOQA = EOQP                              MCG          612     A
MCG                 IFNE   .IWT,0,2                                             MCG          613     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SX6    .IWT                                                    MCG          614     A
MCG                SA6    IWT          IWT = .IWT    */ SET INTERWORD TIME        MCG          615     A
MCG                SX7    FX+FB                                                   MCG          616     A
MCG                MX5    L.CLC/2                                                 MCG          617     A
MCG                PX7    X7                                                      MCG          618     A
MCG                SA7    AXR          AXR = PACK( 0 , FX + FB )                  MCG          619     A
MCG                SX6    FX.LD+FX.XT                                             MCG          620     A
MCG                SA6    ALR          ALR = FX.LD                                MCG          621     A
MCG                SX7    B0                                                      MCG          622     A
MCG                MX6    0                                                       MCG          623     A
MCG                                                                               MCG          624     A
MCG                SA7    F.CLC        FOR I = 0 TO L.CL , CLC(I) = 0             MCG          625     A
MCG        PRS1    SA6    A7+B1                                                   MCG          626     A
MCG                LX5    1                                                       MCG          627     A
MCG                NO                                                             MCG          628     A
MCG                SA7    A6+B1                                                   MCG          629     A
MCG                MI     X5,PRS1                                                 MCG          630     A
MCG                                                                               MCG          631     A
MCG                SA5    N.INST       WDS = N.INST                               MCG          632     A
MCG                ZR     X0,PRS2      IF MODE = 0   */ NOT JAM MODE              MCG          633     A
MCG                BX7    -X5                                                     MCG          634     A
MCG                LX5    1            WDS = 2*N.INST                             MCG          635     A
MCG                SA7    EIW          EIW = -N.INST                              MCG          636     A
MCG                ALLOC  OTI,30B      ALLOC OTI,30B                              MCG          637     A
MCG                                                                               MCG          638     A
MCG        PRS2    ALLOC  PIT,X5+MAX.GX     ALLOC( PIT , WDS+MAX.GX )             MCG          639         I
          -FEAT155A
MCG                SA4    =XPARCEL                                                MCG          640         I
          -FEAT155A
MCG        PRS2    ALLOC  PIT,X5+MAX.GX+1   ALLOC( PIT , WDS+MAX.GX+1 )           FEAT155A      87     A
MCG                SA5    =XPARCEL                                                FEAT155A      88         I
          -CCG4
MCG                SA5    =XCG.PC                                                 CCG4          74         I
          -CCGNC
MCG                SA5    =XCC$PC                                                 CCGNC        782     A
MCG                SB5    B6+B1                                                   MCG          641     A
MCG                BX7    X2                                                      MCG          642     A
MCG                SA7    IIP+1        IIP(2) = O.PIT                             MCG          643     A
MCG                SX6    X2+B5                                                   MCG          644     A
MCG                SA6    A7-B1        IIP(1) = O.PIT + OLD(L.PIT) + 1            MCG          645     A
MCG                BX7    X4                                                      MCG          646         I
          -FEAT155A
MCG                BX7    X5                                                      FEAT155A      89     A
MCG                SA7    PC           PC = PARCEL                                MCG          647         I
          -CCG4
MCG                SA7    PC           PC = CG.PC                                 CCG4          75         I
          -CCGNC
MCG                SA7    PC           PC = CC$PC                                 CCGNC        783     A
MCG                                                                               MCG          648     A
MCG       *        SET UP THE ISSUE CANDIDATE LIST BY ISSUING THE *BOS*           MCG          649     A
MCG                                                                               MCG          650     A
MCG                PRNT   JAM,HEAD                                                MCG          651     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SA3    O.TXT                                                  MCG          652     A
MCG                SB7    X3+3        BASE = O.TXT + 3                           MCG          653     A
MCG                SA4    X3                                                     MCG          654     A
MCG                SB3    B0                                                     MCG          655     A
MCG                BX7    X4          [IIP-1] = [O.TXT]  */ SET *BOS* FOR POST   MCG          656     A
MCG                SA7    X6-1                                                   MCG          657     A
MCG                S"PI"  B0                                                     MCG          658           I
          -FEAT155A
MCG                SA4    =XCO.ER                                                FEAT155A      90           I
          -CCGNC
MCG                SA4    =XHO$ER                                                CCGNC        784     A
MCG                NZ     X5,PRS3     IF PARCEL " 0                              FEAT155A      91     A
MCG                ZR     X4,PRS3     IF ER = 0     */ NO OBJ TIME RPV           FEAT155A      92     A
MCG                                                                              FEAT155A      93     A
MCG                MX5    -R1.INP                                                FEAT155A      94     A
MCG                BX7    X5*X7                                                  FEAT155A      95     A
MCG                SX4    200B                                                   FEAT155A      96     A
MCG                SB2    OC.STT                                                 FEAT155A      97     A
MCG                UX5    X7                                                     CCG0086       11     A
MCG                ZR     X5,PRS3     IF LINE NUMBER ZERO                        CCG0086       12     A
MCG                PX7    B2,X7                                                  FEAT155A      98     A
MCG                BX7    X4+X7                                                  FEAT155A      99     A
MCG                SA7    A7+B1       [IIP] = SI(OC.STT,LINE,200B)               FEAT155A     100     A
MCG                SX6    X6+B1       IIP = IIP + 1                              FEAT155A     101     A
MCG                SA6    A6                                                     FEAT155A     102     A
MCG                SX7    B1+B1                                                  FEAT155A     103     A
MCG                SA7    PC          PC = 2                                     FEAT155A     104     A
MCG                                                                              FEAT155A     105     A
MCG       PRS3     S"PI"  B0                                                     FEAT155A     106     A
MCG                SA5    "TB"                                                   MCG          659     A
MCG                RJ     AIL         INITIALIZE THE ISSUE LIST                  MCG          660     A
MCG                                                                              MCG          661           I
          -CCGA006
MCG                RJ     SSP         SET STORE PREDECESSOR COUNT                CCGA006       10     A
MCG                EQ     PRS                                                    CCGA006       11     A
MCG       SSP      SPACE  3,14                                                   CCGA006       12     A
MCG      **        SSP - SET STORE PREDECESSOR COUNT ( N.SP ) FOR A SUBSEQUENCE  CCGA006       13     A
MCG      *                                                                       CCGA006       14     A
MCG      *         ENTRY  (X4) = [DI] OF INSTRUCTION JUST ISSUED                 CCGA006       15     A
MCG                                                                              CCGA006       16     A
MCG       SSP      ROUTINE                                                       CCGA006       17     A
MCG                LX4    -D.USESP                                               MCG          662     A
MCG                MX5    -D.USESL                                               MCG          663     A
MCG                BX2    -X4+X5                                                 MCG          664     A
MCG                MX7    0                                                      MCG          665     A
MCG                SX6    X2+B1                                                  MCG          666     A
MCG                SA6    N.SP        N.SP = -(USES[DI]-1)                       MCG          667     A
MCG                SA7    STL         STL = 0                                    MCG          668     A
MCG                AX3    B1,X2                                                  MCG          669     A
MCG                ZR     X3,PRS      IF N.SP @ 1                                MCG          670           I
          -CCGA006
MCG                ZR     X3,SSP      IF N.SP @ 1                                CCGA006       18     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SX7    -FX.X6                                                MCG          671     A
MCG                SA7    A7          STL = -FX.X6                              MCG          672     A
MCG                EQ     PRS                                                   MCG          673          I
          -CCGA006
MCG                EQ     SSP                                                   CCGA006       19     A
MCG                TTL    MCG - MACHINE CODE GENERATOR - INSTRUCTION SELECTION  MCG          674     A
MCG       SNI      TITLE  SNI - SELECT NEXT INSTRUCTION TO ISSUE                MCG          675     A
MCG      **        SNI - SELECT NEXT INSTRUCTION TO BE POSTED ( ISSUED ) FROM   MCG          676     A
MCG      *         THE *ICL* ( ISSUE CANIDATE LIST ).                           MCG          677     A
MCG      *         SCAN INSTRUCTIONS ON *ICL* FOR ONE THAT IS EXECUTABLE AT THE MCG          678     A
MCG      *         CURRENT TIME, OR ONE WITH THE BEST ISSUE ( T(IS) ) AND       MCG          679     A
MCG      *         EXECUTE TIME ( T(EX) ) ( EARLIEST START TIME ).              MCG          680     A
MCG      *         NOTE - PSEUDO INSTRUCTIONS AND BOUNDARY MARKERS ARE ISSUED   MCG          681     A
MCG      *                FROM *DRR*.                                           MCG          682     A
MCG      *                                                                      MCG          683     A
MCG      *         EXIT   (B3) = REGNO OF ISSUABLE INST, < 0 IF NONE FOUND      MCG          684     A
MCG      *                (B4,B5) = T(EX), T(IS)                                MCG          685     A
MCG      *                (A5) = LI ( ADDRESS OF LINK WORD OF ISSUABLE INSTRUCTION MCG      686     A
MCG      *                (A0) = *TXT* INDEX OF PREVIOUS INSTRUCTION            MCG          687     A
MCG      *                PBI WORDS SET.                                        MCG          688     A
MCG                                                                             MCG          689     A
MCG       SNIA     BSS    1           TEMP TO SAVE FUNIT ORDINAL                MCG          690     A
MCG                                                                             MCG          691     A
MCG       SNI      ROUTINE                                                      MCG          692     A
MCG                                                                             MCG          693     A
MCG       SNI0     SA4    PBI-1                                                 MCG          694     A
MCG                SA3    ICL         CI = ICL                                  MCG          695     A
MCG                S"PI"  B0          PI = 0                                    MCG          696     A
MCG                BX6    X4                                                    MCG          697     A
MCG                SA6    A4+B1       PBI = PB1(FALSE,0,INFINITY,INFINITY)      MCG          698     A
MCG                SB6    X4          TIS[PBI] = INFINITY                       MCG          699     A
MCG                                                                             MCG          700     A
MCG      *         GET INST DESCR, CHECK FOR FIT IN CURRENT WORD                MCG          701     A
MCG                                                                             MCG          702     A
MCG       SNI1     SB4    X3+"TB"     LI = CI + BASE                            MCG          703     A
MCG                SA2    T                                                     MCG          704     A
MCG                SA4    B4-B1       DI = LI - 1  ( DESCR)                     MCG          705     A
MCG                SB5    X2          T(IS) = T                                 MCG          706     A
MCG                LX4    58-D.SZP                                              MCG          707     A
MCG                PL     X4,SNI4     IF SZ[DI] < 2 */ NO POSSIBLE SIZE FAULT   MCG          708     A
MCG                SA3    PC                                                    MCG          709     A
MCG                MX7    -2                                                    MCG          710     A
MCG                BX6    X3+X7                                                 MCG          711     A
MCG                NZ     X6,SNI4     IF PC " 3     */ NOT IN LAST PARCEL OF WORD MCG          712     A
MCG                BX2    X4                                                    MCG          713     A
MCG                LX2    1+D.SZP-D.SRP                                         MCG          714     A
MCG                PL     X2,SNI3     IF ^SR[DI]    */ NOT SIZE REDUCABLE       MCG          715     A
MCG                                                                             MCG          716     A
MCG      *         INSTRUCTION IS SIZE REDUCEABLE, CHECK A-REGISTERS TO SEE IF  MCG          717     A
MCG      *         ADDRESS IS IN ONE OF THEM.                                   MCG          718     A
MCG                                                                             MCG          719     A
MCG                SA2    ARA+7       I = ARA(8)                                MCG          720     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SA3    "DI"-B1     R2 = DI - 1                                    MCG        721    A
MCG                SB2    ARA-1                                                      MCG        722    A
MCG                BX6    X3                                                         MCG        723    A
MCG                SA6    B2+B1       ARA(1) = [R2]                                  MCG        724    A
MCG                                                                                  MCG        725    A
MCG        SNI2    BX7    X3-X2                                                      MCG        726    A
MCG                SA2    A2-B1       I = I - 1                                      MCG        727    A
MCG                NZ     X7,SNI2     IF [I+1] " [R2]                                MCG        728    A
MCG                SB3    A2-B2                                                      MCG        729    A
MCG                ZR     B3,SNI3     IF I = 0    */ MATCH WITH SEARCH TERMINATOR    MCG        730    A
MCG                                                                                  MCG        731    A
MCG                SA2    TRA+RN.A0+B3              */ COMPUTE TRUE ISSUE TIME       MCG        732    A
MCG                SB2    X2                                                         MCG        733    A
MCG                LE     B2,B5,SNI4                                                 MCG        734    A
MCG                SB5    X2                                                         MCG        735    A
MCG                EQ     SNI4                                                       MCG        736    A
MCG                                                                                  MCG        737    A
MCG        SNI3    SA1    IWT               */ SET PENALITY FOR SIZE FAULT           MCG        738    A
MCG                SX2    X1+B1                                                      MCG        739    A
MCG                SB5    X2+B5       T(IS) = T(IS) + IWT + 1                        MCG        740    A
MCG                                                                                  MCG        741    A
MCG      *         SELECT AN AVAILABLE FUNCTIONAL UNIT                               MCG        742    A
MCG                                                                                  MCG        743    A
MCG        SNI4    LX4    2+D.SZP-D.F1P                                              MCG        744    A
MCG                MX6    -D.F1L                                                     MCG        745    A
MCG                BX7    -X6*X4      F = F1[DI]                                     MCG        746    A
MCG                                                                                  MCG        747    A
MCG        #MD     IFGE   .CPU,74                                   ?66/76          MCG        748    A
MCG                SA3    TFU+X7      TFA = TFU(F)                                   MCG        749    A
MCG                SB3    X3                                                         MCG        750    A
MCG                LE     B3,B5,SNI5  IF TFA @ T(IS)                                 MCG        751    A
MCG        #MDA    IFEQ   .CPU,74                                   ?6600           MCG        752    A
MCG                LX4    D.F1P-D.F2P                                                MCG        753    A
MCG                SX1    X7                        */ CHECK SECOND FUNIT            MCG        754    A
MCG                BX7    -X6*X4      F2 = F2[DI]                                    MCG        755    A
MCG                SA2    TFU+X7      TF2 = TFU(F2)                                  MCG        756    A
MCG                LX4    D.F2P-D.F1P                                                MCG        757    A
MCG                SB4    X2                                                         MCG        758    A
MCG                LE     B4,B5,SNI5  IF TF2 @ T(IS)                                 MCG        759    A
MCG                SB5    B4                                                         MCG        760    A
MCG                LE     B4,B3,SNI5  IF TF2 @ TF1                                   MCG        761    A
MCG                BX7    X1                                                         MCG        762    A
MCG        #MDA    ENDIF                                            '6600           MCG        763    A
MCG                SB5    X3          T(IS) = TF1   */ MIN(TFA,TF2)                  MCG        764    A
MCG        #MD     ENDIF                                            '66/76          MCG        765    A
MCG                                                                                  MCG        766    A
MCG      *         IF T(IS) @ TIS[PBI] THEN TRY TO GET A RESULT REG, CALC T(EX)      MCG        767    A
MCG                                                                                  MCG        768    A
MCG        SNI5    GT     B5,B6,SNI6  IF T(IS) > TIS[PBI]                            MCG        769    A
MCG                SA7    SNIA                                                       MCG        770    A
MCG                LX4    D.F1P+59-D.NRP                                             MCG        771    A
MCG                RJ     DRR         CALL DRR( DI , T(IS) , T(EX) , REGNO )         MCG        772    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG              GT     B3,SNI7     IF REGNO > 0  */ INSTRUCTION IS ISSUABLE          MCG        773    A
MCG              ZR     B3,SNI0     IF REGNO = 0  */ A PSEUDO INSTRUCTION             MCG        774    A
MCG                                                                                   MCG        775    A
MCG       SNI6   SA5    "DI"+B1     LI = DI + 1                                       MCG        776    A
MCG              EQ     SNI9                        */ NO RESULT REG AVAIL             MCG        777    A
MCG                                                                                   MCG        778    A
MCG      *       COMPARE THIS INSTRUCTION TIMING WITH THAT OF *PBI*                   MCG        779    A
MCG                                                                                   MCG        780    A
MCG       SNI7   SA5    "DI"+B1     LI = DI + 1                                       MCG        781    A
MCG       #MD    IFEQ   .CPU,74                                        ?6600          MCG        782    A
MCG              GT     B5,B6,SNI9  IF T(IS) > TIS[PBI]                               MCG        783    A
MCG              SA3    PBI                                                           MCG        784    A
MCG              LT     B5,B6,SNI8  IF T(IS) < TIS[PBI]                               MCG        785    A
MCG              LX3    -PB.TEXP                                                      MCG        786    A
MCG              SB2    X3                                                            MCG        787    A
MCG              GE     B4,B2,SNI9  IF T(EX) \ TEX[PBI]                               MCG        788    A
MCG       #MD    ELSE                                                  '66?76         MCG        789    A
MCG              SA4    "DI"                                                          MCG        790    A
MCG              GE     B4,B6,SNI9  IF T(EX) \ TEX[PBI]                               MCG        791    A
MCG       #MD    IFGT   .CPU,74                                        ?7600          MCG        792    A
MCG              LX4    -D.LDP                                                        MCG        793    A
MCG       .D     BIT    D.^DP-D.LDP                                                   MCG        794    A
MCG              SX6    .D+1                                                          MCG        795    A
MCG              SB5    B4          T(IS) = T(EX)                                     MCG        796    A
MCG              BX7    X6*X4                                                         MCG        797    A
MCG              LX4    D.LDP-D.FTP                                                   MCG        798    A
MCG              NZ     X7,SNI8     IF ^D[DI] ! LD[DI]                                MCG        799    A
MCG                                                                                   MCG        800    A
MCG      *       ADJUST T(EX) TO ACCOUNT FOR THE FUNIT TO REGISTER INPUT PATH         MCG        801    A
MCG                                                                                   MCG        802    A
MCG              SA2    T                                                             MCG        803    A
MCG              SA1    TRS                                                           MCG        804    A
MCG              MX6    -D.FTL                                                        MCG        805    A
MCG              BX7    -X6*X4                                                        MCG        806    A
MCG              IX3    X7-X2                                                         MCG        807    A
MCG              SB2    X3+B5                                                         MCG        808    A
MCG              SB4    B4-B1       T(EX) = T(EX) - 1                                 MCG        809    A
MCG              AX1    B2,X1       XSW = SHIFT(TRS, -(T(IS)+FT[DI]-T) )              MCG        810    A
MCG              MX3    1                                                             CCG0196      5    A
MCG              BX1    -X3*X1                                                        CCG0196      6    A
MCG                                                                                   MCG        811    A
MCG       SNI7A  LX1    -1          XSW = SHIFT( XSW , 59 )                           MCG        812    A
MCG              SB4    B4+B1       T(EX) = T(EX) + 1                                 MCG        813    A
MCG              MI     X1,SNI7A    IF XSW < 0    */ RESULT STORED AT THIS TIME       MCG        814    A
MCG              SB5    B4          T(IS) = T(EX)                                     MCG        815    A
MCG              GE     B4,B6,SNI9  IF T(EX) \ TEX[PBI]                               MCG        816    A
MCG       #MD    ENDIF                                                 '7600          MCG        817    A
MCG                                                                                   MCG        818    A
MCG      *       INSTRUCTION BETTER THAN PREVIOUS BEST, SAVE INFO IN *PBI*            MCG        819    A
MCG                                                                                   MCG        820    A
MCG       SNI8   SX4    B5                                                            MCG        821    A
MCG              SA1    SNIA                                                          MCG        822    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SX3    B4                                                    MCG        823    A
MCG                LX4    PB.TISP                                               MCG        824    A
MCG                PX6    B3,X4                                                 MCG        825    A
MCG                SX7    "PI"                                                  MCG        826    A
MCG                LX3    PB.TEXP                                               MCG        827    A
MCG                SB2    X1                                                    MCG        828    A
MCG                SX2    A5-"TB"                                               MCG        829    A
MCG                BX6    X3+X6                                                 MCG        830    A
MCG                SA1    T                                                     MCG        831    A
MCG                LX7    PB.PIP                                                MCG        832    A
MCG                LX2    PB.LIP                                                MCG        833    A
MCG                PX2    B2,X2                                                 MCG        834    A
MCG                SA6    PBI         PBI(0) = PBII(REG,T(EX),T(IS))            MCG        835    A
MCG                IX7    X7+X2                                                 MCG        836    A
MCG                SB2    X1+                                                   MCG        837    A
MCG                SA7    A6+B1       PBI(1) = PBII2(FU,PI,LI)                  MCG        838    A
MCG                                                                             MCG        839    A
MCG                EQ     B4,B2,SNI   IF T(EX) = T  */ EXIT IF THIS INST IS EXEC MCG        840    A
MCG                                                                             MCG        841    A
MCG                SB6    B5+         TIS[PBI] = T(IS)                          MCG        842    A
MCG                                                                             MCG        843    A
MCG       *        ADVANCE TO NEXT INSTRUCTION ON CHAIN                         MCG        844    A
MCG                                                                             MCG        845    A
MCG        SNI9    S"PI"  A5-"TB"     PI = LI - BASE                            MCG        846    A
MCG                LX5    -I.LINKP                                              MCG        847    A
MCG                SX3    X5          CI = LINK[LI]                             MCG        848    A
MCG        #MD     IFGE   .CPU,74                                   ?66/76     MCG        849    A
MCG                SA1    PBI                                                   MCG        850    A
MCG                ZR     X3,SNI10    IF CI = 0                                 MCG        851    A
MCG                                                                             MCG        852    A
MCG       *        ONLY CONSIDER INSTRUCTIONS THAT ARE "LATE" SO WE DO NOT CLOG MCG        853    A
MCG       *        UP THE REGISTERS WITH RESULTS THAT CANNOT BE USED.           MCG        854    A
MCG       *        I.E. DO NOT TRY TO CODE THE SEQUENCE IN A TIME LESS THAN THE MCG        855    A
MCG       *        CRITICAL PATH TIME ( AN IMPOSSABILITY ).                     MCG        856    A
MCG                                                                             MCG        857    A
MCG                SA5    "TB"+X3     LI = BASE + CI                            MCG        858    A
MCG                UX7    B3,X1                                                 MCG        859    A
MCG                MI     B3,SNI1     IF REG[PBI] < 0 */NO ISSUABLE INST FOUND  MCG        860    A
MCG                SA2    T                                                     MCG        861    A
MCG                MX6    -I.LSTL                                               MCG        862    A
MCG                LX5    -I.LSTP                                               MCG        863    A
MCG                BX7    -X6*X5                                                MCG        864    A
MCG                IX6    X2-X7                                                 MCG        865    A
MCG        SNI.P   SX6    X4+DELTAT                                             MCG        866    A
MCG                PL     X6,SNI1     IF IF LST[LI] @ T+DT  */ INST IS *LATE*   MCG        867    A
MCG                                                                             MCG        868    A
MCG        SNI10   BSS    0                                                     MCG        869    A
MCG        #MD     ELSE                                             ?66/76     MCG        870    A
MCG                NZ     X3,SNI1     IF CI " 0                                 MCG        871    A
MCG                                                                             MCG        872    A
MCG                SA1    PBI                                                   MCG        873    A
MCG        #MD     ENDIF                                                        MCG        874    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SA2    A1+B1                                                  MCG        875    A
MCG                UX3    B3,X1       REGNO = REG[PBI]                           MCG        876    A
MCG                LX2    -PB.LIP                                                MCG        877    A
MCG                MI     B3,SNI      IF REGNO < 0  */ NO ISSUABLE INST FOUND    MCG        878    A
MCG                                                                              MCG        879    A
MCG                SA5    X2+"TB"     LI = LI[PBI] + BASE                        MCG        880    A
MCG                LX2    PB.LIP-PB.PIP                                          MCG        881    A
MCG                S"PI"  X2          PI = PI[PBI]                               MCG        882    A
MCG                LX1    -PB.TISP                                               MCG        883    A
MCG                SB5    X1          T(IS) = TIS[PBI]                           MCG        884    A
MCG                LX1    PB.TISP-PB.TEXP                                        MCG        885    A
MCG                SB4    X1          T(EX) = TEX[PBI]                           MCG        886    A
MCG                EQ     SNI                                                    MCG        887    A
MCG                SPACE  2                                                      MCG        888    A
MCG                IFGE   .CPU,74,2                                              MCG        889    A
MCG       SNIB     SX6    X4+0        SKELTON TO PLUG *SNI.P*                    MCG        890    A
MCG                PL     X4,SNI1                                                MCG        891    A
MCG       DRR      TITLE  DRR - DETERMINE RESULT REGISTER / T(EX)                MCG        892    A
MCG      **        DRR - DETERMINE RESULT REGISTER / T(EX) OF INSTRUCTION        MCG        893    A
MCG      *                                                                       MCG        894    A
MCG      *         ENTRY  (A4) = INSTRUCTION DESCRIPTOR ADDRESS ( DI )           MCG        895    A
MCG      *                (X4) = SHIFT([DI],59-D.NRP)                            MCG        896    A
MCG      *                                                                       MCG        897    A
MCG      *         EXIT   (B5) = T(IS) UPDATED                                   MCG        898    A
MCG      *                (B4) = T(EX)                                           MCG        899    A
MCG      *                (B3) = ACT/REGNO, ACT = 0, THEN ISSUED PSEUDO INSTRUCTION MCG    900    A
MCG      *                                  ACT < 0, IF NO RESULT REG AVAIL     MCG        901    A
MCG      *                                  ACT > 0, REGNO = ACT                MCG        902    A
MCG      *                                                                       MCG        903    A
MCG      *         CALLS - CRU, FXR , PPI                                        MCG        904    A
MCG      *         PRESERVES A0, A4, X0, B6, B7                                  MCG        905    A
MCG                                                                              MCG        906    A
MCG       DRR      ROUTINE                                                       MCG        907    A
MCG                NX6    B3,X4                                                  MCG        908    A
MCG                SB4    B5+         T(EX) = T(IS)                              MCG        909    A
MCG                LX4    1+D.NRP-D.TYP            */ [DI] POS FOR TYPE EXTRACT  MCG        910    A
MCG                JP     B3+DRR1     JUMP( NORMC( [X4] ) )                      MCG        911    A
MCG                                                                              MCG        912    A
MCG       DRR1     BSS    0                                                      MCG        913    A
MCG                QUAL   DRR                                                    MCG        914    A
MCG                                                                              MCG        915    A
MCG                SA5    "DI"+2      R1RS = DI + 2                              MCG        916    A
MCG                EQ     RJRS                                                   MCG        917    A
MCG                                                                              MCG        918    A
MCG                EQ     PS                                                     MCG        919    A
MCG                                                                              MCG        920    A
MCG                SA5    "DI"+2      R1RS = DI + 2                              MCG        921    A
MCG                EQ     PRS                                                    MCG        922    A
MCG                                                                              MCG        923    A
MCG                SA5    "DI"-2      R1 = DI - 2                                CCG3       117    A
MCG                EQ     ECJ         EC JUMP                                    MCG        924    A
MCG                                                                              MCG        925    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                EQ     LVL2                                                  MCG        926     A
MCG                                                                             MCG        927     A
MCG                EQ     LD                                                    MCG        928     A
MCG                                                                             MCG        929     A
MCG                SA5    "DI"-2      R1 = DI - 2                               MCG        930     A
MCG                EQ     ST                                                    MCG        931     A
MCG                                                                             MCG        932     A
MCG                EQ     JP                                                    MCG        933     A
MCG                                                                             MCG        934     A
MCG                SA5    "DI"-2      R1 = DI - 2                               MCG        935     A
MCG                EQ     XMT                                                   MCG        936     A
MCG                                                                             MCG        937     A
MCG                SA5    "DI"-B1     R2 = DI - 1                               MCG        938     A
MCG                EQ     KS                                                    MCG        939     A
MCG                                                                             MCG        940     A
MCG                SA5    "DI"-2      R1 = DI - 2                               MCG        941     A
MCG                EQ     PI                        */ PSEUDO INSTRUCTION       MCG        942     A
MCG                                                                             MCG        943     A
MCG                EQ     MUC                                                   MCG        944     A
MCG                                                                             MCG        945     A
MCG       *        EQ     SCR         ANY X REGISTER                            MCG        946     A
MCG        SCR     EJECT                                                        MCG        947     A
MCG       *        SCR - FIND AN X-REGISTER                                     MCG        948     A
MCG                                                                             MCG        949     A
MCG        SCR     SA3    STL                                                   MCG        950     A
MCG                SX5    FX.XT+FX.SCR+X3   CLASS = SCR & ^STL                  MCG        951     A
MCG                                                                             MCG        952     A
MCG        SCR1    FXR    X5,DRR2     REGNO = FXR( ^STL )                       MCG        953     A
MCG                                                                             MCG        954     A
MCG       *        NO REGISTER AVAILABLE, CHECK STORE LOCKOUT FLAG               MCG        955     A
MCG                                                                             MCG        956     A
MCG                SA5    AXR         CLASS = AXR                               MCG        957     A
MCG                SA3    STL                                                   MCG        958     A
MCG                BX7    X3+X5                                                 MCG        959     A
MCG                ZR     X7,SCR1     IF AXR & STL " 0  */ X6 LOCKOUT AND AVAIL MCG        960     A
MCG                ZR     X3,DRR      IF STL = 0    */ NO *ST* LOCKOUT          MCG        961     A
MCG                SA3    L.OTI                                                 MCG        962     A
MCG                ZR     X3,DRR      IF L.OTI = 0  */ NOT JAM MODE             MCG        963     A
MCG                                                                             MCG        964     A
MCG                FXR    ST,DRR2     REGNO = FXR( ST )                         MCG        965     A
MCG                EQ     DRR                                                   MCG        966     A
MCG        MUC     SPACE  2                                                     MCG        967     A
MCG       *        MUC - MULTI USE COMPUTATION                                  MCG        968     A
MCG                                                                             MCG        969     A
MCG        MUC     FXR    ST,DRR2     REGNO = FXR( ST )                         MCG        970     A
MCG                SX5    FX.XT+FX.SCR      CLASS = SCR                         MCG        971     A
MCG                EQ     SCR1                                                  MCG        972     A
MCG        RJRS    SPACE  3,14                                                  MCG        973     A
MCG       **       INSTRUCTION PRECEDES AN RJ *RS*  ( UP / NR )                 MCG        974     A
MCG                                                                             MCG        975     A
MCG        RJRS    MX2    -SO.REGL                                              MCG        976         I
          -FCC0641

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                LX5    -R1.SOP                                                MCG          977          I
          -FCC0641
MCG                SB3    -B1         ACT = -1                                   MCG          978          I
          -FCC0641
MCG                BX6    -X2*X5      REGNO = SOREG[R1RS]                        MCG          979          I
          -FCC0641
MCG                SA1    RVT+X6                                                 MCG          980          I
          -FCC0641
MCG                MX7    -I.USESL                                               MCG          981          I
          -FCC0641
MCG                SA2    "TB"+X1     LP = TB + RVT(REGNO)                       MCG          982          I
          -FCC0641
MCG                LX7    I.USESP                                                MCG          983          I
          -FCC0641
MCG                BX3    -X7*X2                                                 MCG          984          I
          -FCC0641
MCG                NZ     X3,DRR      IF USES[LP] " 0  */ B-REG IS IN USE        MCG          985          I
          -FCC0641
MCG       RJRS     RJ     FBR         REGNO = FBR( R1RS )                        FCC0641       17     A
MCG                MI     B3,DRR      IF ACT < 0    */ NO B-REG FOUND            FCC0641       18     A
MCG                SA2    TRA+X6                    */ ADJUST ISSUE TIME         MCG          986     A
MCG                SB2    X2                                                     MCG          987     A
MCG                LE     B2,B5,RJRS1 T(IS) = MAX( T(IS) , TRA(REGNO) )          MCG          988     A
MCG                SB4    X2                                                     MCG          989     A
MCG                SB5    B2                                                     MCG          990     A
MCG                                                                             MCG          991     A
MCG       RJRS1    BX7    X4                                                     MCG          992     A
MCG                LX7    D.TYP-1-D.PSP                                          MCG          993     A
MCG                PL     X7,SCR      IF ^PS[DI]                                 MCG          994     A
MCG       PS       SPACE  3,8                                                    MCG          995     A
MCG      *         INSTRUCTION PRECEDES A STORE                                  MCG          996     A
MCG                                                                             MCG          997     A
MCG       PS       FXR    ST,DRR2     REGNO = FXR( ST )                          MCG          998     A
MCG                EQ     DRR                                                    MCG          999     A
MCG       PRS      SPACE  3,14                                                   MCG         1000     A
MCG      *         INSTRUCTION PRECEDES A *RS* ( REG STORE )                     MCG         1001     A
MCG                                                                             MCG         1002     A
MCG       PRS      MX2    -SO.REGL                                               MCG         1003     A
MCG                LX5    -R1.SOP                                                MCG         1004     A
MCG                BX6    -X2*X5      REGNO = SOREG[R1RS]                        MCG         1005     A
MCG                SB3    X6                                                     MCG         1006     A
MCG                RJ     CRU         ACT = CRU( REGNO )                         MCG         1007     A
MCG                MI     B3,DRR      IF ACT < 0    */ REGNO NOT AVAIL NOW       MCG         1008     A
MCG                                                                             MCG         1009     A
MCG       #MD      IFEQ   .CPU,74                                   ?6600       MCG         1010          I
          -FCC0881
MCG                BX7    X4                                                     MCG         1011          I
          -FCC0881
MCG                LX7    D.TYP-1-D.LDP                                          MCG         1012          I
          -FCC0881
MCG                PL     X7,DRR2     IF ^LD[DI]                                 MCG         1013          I
          -FCC0881

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                EQ     LD1                                                  MCG        1014        I
          -FCC0881
MCG        #MD     ELSE                                   '6600                 MCG        1015        I
          -FCC0881
MCG                EQ     DRR2                                                 MCG        1016        I
          -FCC0881
MCG        #MD     ENDIF                                  '7600                 MCG        1017        I
          -FCC0881
MCG                BX7    X4                                                   FCC0881       8     A
MCG                LX7    D.TYP-1-D.XMTP                                       FCC0881       9     A
MCG                MI     X7,PRS1      IF XMT[DI]                              FCC0881      10     A
MCG                                                                            FCC0881      11     A
MCG                IFEQ   .CPU,74,2                                   ?6600   FCC0881      12     A
MCG                LX7    D.XMTP-D.LDP                                         FCC0881      13     A
MCG                MI     X7,LD1       IF LD[DI]                               FCC0881      14     A
MCG                EQ     DRR2                                                 FCC0881      15     A
MCG                                                                            FCC0881      16     A
MCG        PRS1    SA5    "DI"-2       R1 = DI - 2                             FCC0881      17     A
MCG                LX5    -R1.RJP                                              FCC0881      18     A
MCG                BX7    "RN"X5                                               FCC0881      19     A
MCG                SA2    "TB"+X7      LJ = BASE + RJ[R1]                      FCC0881      20     A
MCG                LX2    -I.REGP                                              FCC0881      21     A
MCG                SB2    X2                                                   FCC0881      22     A
MCG                NE     B2,B3,DRR2   IF REG[LJ] " REGNO                      FCC0881      23     A
MCG                                                                            FCC0881      24     A
MCG                LX5    R1.RJP                                               FCC0881      25     A
MCG                LX2    I.REGP             (X2) = LINK WORD OF PRED          CCGA006      20     A
MCG                BX7    "RN"X5       RI = RN[R1]                             FCC0881      26     A
MCG                EQ     XMT1                                                 FCC0881      27     A
MCG        LVL2    SPACE  3,10                                                 MCG        1018     A
MCG       *        LEVEL 2 LD/ST  ( DRL/DWL )                                  MCG        1019     A
MCG                                                                            MCG        1020     A
MCG        LVL2    LX6    D.L2P-47+59-D.STP                                    MCG        1021     A
MCG                PL     X6,SCR       IF ^ST[DI]                              MCG        1022     A
MCG                SA5    "DI"-2       R1 = DI - 2                             MCG        1023     A
MCG                EQ     ST                                                   MCG        1024     A
MCG        LD      SPACE  3,10                                                 MCG        1025     A
MCG       *        LD - INSTRUCTION IS A LOAD                                  MCG        1026     A
MCG                                                                            MCG        1027     A
MCG        LD      SA5    ALR                                                  MCG        1028     A
MCG                FXR    X5,LD1,DRR   REGNO = FXR( ALR )                      MCG        1029     A
MCG        LD1     BSS    0                                                    MCG        1030     A
MCG        #MD     IFEQ   .CPU,74                                     ?6600   MCG        1031     A
MCG                                                                            MCG        1032     A
MCG       *        INSTRUCTION CANNOT ISSUE UNTIL LAST STORE IS COMPLETE       MCG        1033     A
MCG                                                                            MCG        1034     A
MCG                SA2    LST                                                  MCG        1035     A
MCG                SB2    X2                                                   MCG        1036     A
MCG                LE     B2,B5,DRR2   T(IS) = MAX( T(IS) , LST )              MCG        1037     A
MCG                SB5    X2                                                   MCG        1038     A
MCG                SB4    X2                                                   MCG        1039     A
MCG        #MD     ENDIF                                          '6600       MCG        1040     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                EQ     DRR2                                                    MCG        1041     A
MCG        ST      SPACE  3,10                                                    MCG        1042     A
MCG       *        STORE INSTRUCTION                                              MCG        1043     A
MCG                                                                               MCG        1044     A
MCG        ST      BX6    "RN"X5      RI = RN[R1]                                 MCG        1045     A
MCG                SA3    "TB"+X6     LP = BASE + RI                              MCG        1046     A
MCG                LX3    -I.REGP                                                 MCG        1047     A
MCG                SB3    X3          REGNO = REG[LP]                             MCG        1048     A
MCG                SA2    TRA+X3                                                  MCG        1049     A
MCG                SB2    X2                                                      MCG        1050     A
MCG        #MD     IFEQ   .CPU,74                                    ?6600       MCG        1051     A
MCG                LE     B2,B5,ST1   T(IS) = MAX( T(IS) , TRA(REGNO) )           MCG        1052     A
MCG                SB5    X2                                                      MCG        1053     A
MCG                SB4    X2                                                      MCG        1054     A
MCG        ST1     SA1    LST                                                     MCG        1055     A
MCG                SA2    LLT                                                     MCG        1056     A
MCG                IX7    X1-X2                                                   MCG        1057     A
MCG                MI     X7,ST2      TLS = MAX( LST , LLT )                      MCG        1058     A
MCG                BX2    X1                                                      MCG        1059     A
MCG        ST2     SB2    X2                                                      MCG        1060     A
MCG        #MD     ENDIF                                             '6600       MCG        1061     A
MCG                LE     B2,B5,DRR2  T(IS/EX) = MAX( T(IS/EX) , TLS )            MCG        1062     A
MCG                SB5    X2                                                      MCG        1063     A
MCG                SB4    X2                                                      MCG        1064     A
MCG                EQ     DRR2                                                    MCG        1065     A
MCG        JP      SPACE  2                                                       MCG        1066     A
MCG       *        JUMP INSTRUCTION                                               MCG        1067          I
          -CCG3
MCG                                                                               MCG        1068          I
          -CCG3
MCG        ECJ     SX6    2                                                       MCG        1069          I
          -CCG3
MCG                SA6    PC          PC = 2                                      MCG        1070          I
          -CCG3
MCG       *        CONDITIONAL RJ ( RJXJ )                                        CCG3        118     A
MCG                                                                               CCG3        119     A
MCG        ECJ     SA2    PC                                                      CCG3        120     A
MCG                ZR     X2,ECJ1     IF PC = 0     */ AT BEGINNING OF A WORD     CCG3        121     A
MCG                SX2    X2-3                                                    CCG3        122     A
MCG                ZR     X2,ECJ1     IF PC = 3     */ SNI BUMPED T(IS)           CCG3        123     A
MCG                SA1    IWT                                                     CCG3        124     A
MCG                IX2    X1-X2                                                   CCG3        125     A
MCG                SB5    B5+X2       T(SI) = T(IS) + IWT + 3-PC                  CCG3        126     A
MCG        ECJ1    SB4    B5          T(EX) = T(IS)                               CCG3        127     A
MCG                BX6    "RN"X5                                                  CCG3        128     A
MCG                SA3    "TB"+X6     LP = BASE + RI[R1]                          CCG3        129     A
MCG                LX3    -I.REGP                                                 CCG3        130     A
MCG                SB3    X3          REGNO = REG[LP]                             CCG3        131     A
MCG                SA2    TRA+X3                                                  CCG3        132     A
MCG                SB2    X2+                                                     CCG3        133     A
MCG                LE     B2,B4,DRR   T(EX) = MAX( T(EX) , TRA(REGNO) )           CCG3        134     A
MCG                SB4    B2+                                                     CCG3        135     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                EQ     DRR                                                  CCG3        136     A
MCG                                                                             CCG3        137     A
MCG       *        JUMPS ( JPX, JPBB, JIN, UJP, RJX )                           CCG3        138     A
MCG                                                                             MCG        1071     A
MCG        JP      RJ     PJI         PROCESS THE JUMP                          MCG        1072     A
MCG                SB3    B0          REGNO = 0                                 MCG        1073     A
MCG                EQ     DRR                                                  MCG        1074     A
MCG        XMT     SPACE  2                                                     MCG        1075     A
MCG       *        XMT/SXT INSTRUCTIONS                                         MCG        1076     A
MCG                                                                             MCG        1077     A
MCG        XMT     BX7    "RN"X5      RI = RN[R1]                               MCG        1078     A
MCG                LX5    -R1.RJP                                               MCG        1079     A
MCG                BX6    "RN"X5      RJ = RJ[R1]                               MCG        1080     A
MCG                SA3    "TB"+X6     LJ = BASE + RJ                            MCG        1081          I
          -CCGA006
MCG                MX1    -I.USESL+1                                            MCG        1082          I
          -CCGA006
MCG                SB3    X3          REGNO = REG[LJ]                           MCG        1083          I
          -CCGA006
MCG                LX3    -I.USESP-1                                            MCG        1084          I
          -CCGA006
MCG                BX2    -X1*X3                                                MCG        1085          I
          -CCGA006
MCG                NZ     X2,SCR      IF USES[LJ] " 1                           MCG        1086          I
          -CCGA006
MCG                SA2    "TB"+X6     LJ = BASE + RJ                            CCGA006      21     A
MCG                MX1    -I.USESL+1                                            CCGA006      22     A
MCG                SB3    X2          REGNO = REG[LJ]                           CCGA006      23     A
MCG                LX1    I.USESP+1                                             CCGA006      24     A
MCG                BX3    -X1*X2                                                CCGA006      25     A
MCG                NZ     X3,SCR      IF USES[LJ] " 1                           CCGA006      26     A
MCG                SA1    RVT+B3                                                MCG        1087     A
MCG                MI     X1,SCR      IF LOCK[RVT(REGNO)]                       MCG        1088     A
MCG                                                                             MCG        1089     A
MCG       *        ISSUE THE INSTRUCTION                                        MCG        1090     A
MCG                                                                             MCG        1091     A
MCG                SA7    A1          RVT(REGNO) = RI                           MCG        1092          I
          -FCC0881
MCG        XMT1    SA7    RVT+B3      RVT(REGNO) = RI                           FCC0881      28     A
MCG                SX3    B1                                                    CCGA006      27     A
MCG                LX3    I.USESP                                               CCGA006      28     A
MCG                IX6    X2-X3       USES[LJ] = USES[LJ] - 1                   CCGA006      29     A
MCG                SA6    A2                                                    CCGA006      30     A
MCG                SA5    "DI"+B1     LI = DI + 1                               MCG        1093     A
MCG                RJ     AIL         CALL AIL( PI , LI , REGNO )               MCG        1094     A
MCG                PRINT  II,(* XMT X*O1*   OMITTED*),B3                        FCC0881      29     A
MCG                SB3    B0          ACT = 0                                   MCG        1095     A
MCG                EQ     DRR                                                  MCG        1096     A
MCG        KS      SPACE  3,10                                                  MCG        1097     A
MCG       *        CONSTANT SHIFT INSTRUCTION                                   MCG        1098     A
MCG                                                                             MCG        1099     A
MCG        KS      LX5    -IH.RFP                                               MCG        1100     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SA3    "TB"+X5      LJ = BASE + RF[R2]                          MCG        1101     A
MCG                LX3    -I.REGP                                                  MCG        1102     A
MCG                SA2    TRA+X3                                                   MCG        1103     A
MCG                SB3    X3           REGNO = REG[LJ]                             MCG        1104     A
MCG                SB2    X2                                                       MCG        1105     A
MCG                LE     B2,B5,DRR    T(IS/EX) = MAX( T(IS/EX) , TRA(REGNO) )     MCG        1106     A
MCG                SB4    X2                                                       MCG        1107     A
MCG                SB5    X2                                                       MCG        1108     A
MCG                EQ     DRR                                                      MCG        1109     A
MCG         PI     SPACE  3,8                                                      MCG        1110     A
MCG        *       PSEUDO INSTRUCTIONS ( EOQ, BOS, EOS, DAR, DEF, RS, LAB, ENT )  MCG        1111     A
MCG                                                                                MCG        1112     A
MCG         PI     RJ     PPI                                                      MCG        1113     A
MCG                SB3    B0           ACT = 0                                     MCG        1114     A
MCG                EQ     DRR                                                      MCG        1115     A
MCG         GOT    SPACE  2                                                        MCG        1116     A
MCG         #MD    IFLT   .CPU,74                                    '6400        MCG        1117     A
MCG         DRR2   EQU    DRR                                                      MCG        1118     A
MCG         #MD    ELSE                                              '6400        MCG        1119     A
MCG                                                                                MCG        1120     A
MCG        *       GET OPERAND AVAILABLE TIMES AND ADJUST T(EX) ( TYPES I & III )  MCG        1121     A
MCG                                                                                MCG        1122     A
MCG         DRR2   MX1    -D.TYL                                                  MCG        1123     A
MCG                SA5    "DI"-2       R1 = DI - 2                                 MCG        1124     A
MCG                BX7    -X1*X4                                                  MCG        1125     A
MCG                NZ     X7,DRR5      IF TYPE[DI] " I                             MCG        1126     A
MCG                                                                                MCG        1127     A
MCG                AX5    R1.RIL                                                  MCG        1128     A
MCG                BX7    "RN"X5       R = RK[R1]                                  MCG        1129     A
MCG                SA2    "TB"+X7                                                  MCG        1130     A
MCG                AX5    R1.RIL                                                  MCG        1131     A
MCG                LX2    -I.REGP      REGNO = REG[BASE+RN]                        MCG        1132     A
MCG                SA1    TRA+X2                                                   MCG        1133     A
MCG                SB2    X1                                                       MCG        1134     A
MCG                LE     B2,B4,DRR3   T(EX) = MAX( T(EX) , TRA(REGNO) )           MCG        1135     A
MCG                SB4    X1                                                       MCG        1136     A
MCG         DRR3   BX7    "RN"X5       R = RJ[R1]                                  MCG        1137     A
MCG                SA2    "TB"+X7                                                  MCG        1138     A
MCG                LX2    -I.REGP                                                  MCG        1139     A
MCG                SA1    TRA+X2                                                   MCG        1140     A
MCG                SB2    X1                                                       MCG        1141     A
MCG                LE     B2,B4,DRR    T(EX) = MAX( T(EX) , TRA(REGNO) )           MCG        1142     A
MCG                SB4    X1                                                       MCG        1143     A
MCG                EQ     DRR                                                      MCG        1144     A
MCG                                                                                MCG        1145     A
MCG        *       PROCESS TYPE III INSTRUCTION, *RI* IS ALWAYS A DEFINATION      MCG        1146     A
MCG                                                                                MCG        1147     A
MCG         DRR5   AX7    1                                                       MCG        1148     A
MCG                ZR     X7,DRR       IF TYPE[DI] = II  */ HAS NO OPERANDS        MCG        1149     A
MCG                SA2    "DI"-B1      R2 = DI - 1                                 MCG        1150     A
MCG                LX2    -IH.RFP                                                 MCG        1151     A
MCG                BX3    "RN"X2       RF = RF[R2]                                 MCG        1152     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                ZR     X3,DRR      IF RF = 0                                    MCG          1153     A
MCG                SA2    "TB"+X3                                                  MCG          1154     A
MCG                LX2    -I.REGP                                                  MCG          1155     A
MCG                SA3    TRA+X2                                                   MCG          1156     A
MCG                SB2    X3                                                       MCG          1157     A
MCG                LE     B2,B4,DRR   T(EX) = MAX( T(EX) , TRA(REGNO) )            MCG          1158     A
MCG                SB4    B2                                                       MCG          1159     A
MCG                EQ     DRR                                                      MCG          1160     A
MCG                                                                                MCG          1161     A
MCG        #MD     ENDIF                                                           MCG          1162     A
MCG                QUAL                                                            MCG          1163     A
MCG        CRU     TITLE  CRU - CHECK REMAINING USES                               MCG          1164     A
MCG       **       CRU - CHECK REMAINING USES OF SPECIFIED REGISTER                MCG          1165     A
MCG       *               TO SEE IF INSTRUCTION CAN BE ISSUED NOW                 MCG          1166     A
MCG       *                                                                       MCG          1167     A
MCG       *        ENTRY  (B3) = REGNO                                             MCG          1168     A
MCG       *               (A4,X4) = DI, SHIFT([DI], -D.TYP )                      MCG          1169     A
MCG       *                                                                       MCG          1170     A
MCG       *        EXIT   (B3) = REGNO, IF AVAIL, ELSE < 0                         MCG          1171     A
MCG       *               B4, B5 UPDATED                                          MCG          1172     A
MCG                                                                                MCG          1173     A
MCG        CRU0    SA2    TRA+B2                                                   MCG          1174     A
MCG                SB3    B2          REGNO = SAVE                                 MCG          1175     A
MCG                SB2    X2                                                       MCG          1176     A
MCG                LE     B2,B5,CRU   T(IS/EX) = MAX( T(IS/EX) , TRA(REGNO) )      MCG          1177     A
MCG                SB4    X2                                                       MCG          1178     A
MCG                SB5    X2                                                       MCG          1179     A
MCG                                                                                MCG          1180     A
MCG        CRU     ROUTINE                                                         MCG          1181     A
MCG                SA1    RVT+B3      R = RVT(REGNO)                               MCG          1182          I
          -FCC1056
MCG                SX6    B3-26B                                                   FCC1056         9     A
MCG                MI     X6,CRU1     IF REGNO < 26B  */ ^ A ST-REGISTER           FCC1056        10     A
MCG                SA1    L.OTI                                                    FCC1056        11     A
MCG                NZ     X1,CRU1     IF L.OTI " 0  */ IN JAM MODE                 FCC1056        12     A
MCG                SA2    N.SP                                                     FCC1056        13     A
MCG                SX6    X2-1                                                     FCC1056        14     A
MCG                ZR     X6,CRU1     IF N.SP = 1   */ ALL STORES ISSUED           FCC1056        15     A
MCG                SX1    B1                                                       FCC1056        16     A
MCG                SX2    B3                                                       FCC1056        17     A
MCG                BX3    X2-X1       OSR = XOR(REGNO,1)                           FCC1056        18     A
MCG                SA1    RVT+X3                                                   FCC1056        19     A
MCG                SB2    "TB"-B1                                                  FCC1056        20     A
MCG                SA2    B2+X1       DOS = BASE-1 + RI[RVT(OSR)]                  FCC1056        21     A
MCG                LX2    59-D.PRSP                                                FCC1056        22     A
MCG                PL     X2,CRU1     IF ^PRS[DOS]                                 FCC1056        23     A
MCG                                                                                FCC1056        24     A
MCG                SB3    -B1         REGNO = -1    */ INDICATE REG NOT AVAIL      FCC1056        25     A
MCG                EQ     CRU                                                      FCC1056        26     A
MCG                                                                                FCC1056        27     A
MCG        CRU1    SA1    RVT+B3      R = RVT(REGNO)                               FCC1056        28     A
MCG                SB2    B3          SAVE = REGNO                                 MCG          1183     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SB3    -B1         REGNO = -1                                  MCG          1184     A
MCG                SA2    AXR                                                     MCG          1185     A
MCG                SA3    RBV+B2                                                  MCG          1186     A
MCG                BX6    X2*X3                                                   MCG          1187     A
MCG                NZ     X6,CRU0     IF AXR & RBV(REGNO) " 0 */ REG IS AVAIL     MCG          1188     A
MCG                SA3    "TB"+X1     LP = BASE + R                               MCG          1189     A
MCG                MX2    -I.USESL                                                MCG          1190     A
MCG                LX3    -I.USESP                                                MCG          1191     A
MCG                SX1    X1                        */ REMOVE LOCK BIT            MCG          1192     A
MCG                BX2    -X2*X3      U = USES[LP]                                MCG          1193     A
MCG                MX6    -D.TYL                                                  MCG          1194     A
MCG                ZR     X2,CRU0     IF U = 0                                    MCG          1195          I
          -FCC1555X
MCG                ZR     X2,CRU4     IF U = 0                                    FCC1555X        4     A
MCG                SX5    X2-2                                                    MCG          1196     A
MCG                BX7    -X6*X4                                                  MCG          1197     A
MCG                MI     X5,CRU1     IF U = 1                                    MCG          1198          I
          -FCC1056
MCG                MI     X5,CRU2     IF U = 1                                    FCC1056        29     A
MCG                NZ     X5,CRU      IF U > 2                                    MCG          1199     A
MCG                                                                               MCG          1200     A
MCG       *        CHECK 2 USES CASE                                              MCG          1201     A
MCG                                                                               MCG          1202     A
MCG                NZ     X7,CRU      IF TYPE[DI] " I                             MCG          1203     A
MCG                SA5    "DI"-2      R1 = DI - 2                                 MCG          1204     A
MCG                LX5    -R1.RJP                                                 MCG          1205     A
MCG                BX7    "RN"X5                                                  MCG          1206     A
MCG                LX5    R1.RJP-R1.RKP                                           MCG          1207     A
MCG                IX2    X7-X1                                                   MCG          1208     A
MCG                NZ     X2,CRU      IF RJ[R1] " R                               MCG          1209     A
MCG                BX5    "RN"X5                                                  MCG          1210     A
MCG                IX2    X7-X5                                                   MCG          1211     A
MCG                ZR     X2,CRU0     IF RJ[R1] = RK[R1]                          MCG          1212     A
MCG                EQ     CRU                                                     MCG          1213     A
MCG                                                                               MCG          1214     A
MCG        CRU1    LX7    -1                                                      MCG          1215          I
          -FCC1056
MCG        CRU2    LX7    -1                                                      FCC1056        30     A
MCG                MI     X7,CRU      IF TYPE[DI] = II ! IV                       MCG          1216     A
MCG                ZR     X7,CRU2     IF TYPE[DI] = I                             MCG          1217          I
          -FCC1056
MCG                ZR     X7,CRU3     IF TYPE[DI] = I                             FCC1056        31     A
MCG        CRU     SPACE  1,10                                                    MCG          1218     A
MCG       *        CHECK TYPE III INSTRUCTION                                     MCG          1219     A
MCG                                                                               MCG          1220     A
MCG                SA5    "DI"-B1     R2 = DI - 1                                 MCG          1221     A
MCG                LX5    -IH.RFP                                                 MCG          1222     A
MCG                BX7    "RN"X5                                                  MCG          1223     A
MCG                IX2    X7-X1                                                   MCG          1224     A
MCG                ZR     X2,CRU0     IF RF[R2] = R                               MCG          1225     A
MCG                EQ     CRU                                                     MCG          1226     A
MCG                                                                               MCG          1227     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG       CRU2     SA5    "DI"-2      R1 = DI - 2                                   MCG         1228          I
          -FCC1056
MCG       CRU3     SA5    "DI"-2      R1 = DI - 2                                   FCC1056       32     A
MCG                LX5    -R1.RJP                                                   MCG         1229     A
MCG                BX7    "RN"X5                                                    MCG         1230     A
MCG                IX2    X1-X7                                                     MCG         1231     A
MCG                ZR     X2,CRU0     IF RJ[R1] = R                                 MCG         1232     A
MCG                LX5    R1.RJP-R1.RKP                                             MCG         1233     A
MCG                BX7    "RN"X5                                                    MCG         1234     A
MCG                IX2    X1-X7                                                     MCG         1235     A
MCG                ZR     X2,CRU0     IF RK[R1] = R                                 MCG         1236     A
MCG                EQ     CRU                                                       MCG         1237     A
MCG                                                                                FCC1555X       5     A
MCG       *        USES = 0, R-NUM IN REG MUST BE RESULT OF A FULL LOCK,             FCC1555X       6     A
MCG       *        IF THIS INSTRUCTION *PRS* A TEMP LOCK, THEN TRY TO ASSIGN         FCC1555X       7     A
MCG       *        ANOTHER B-REGISTER.                                               FCC1555X       8     A
MCG                                                                                FCC1555X       9     A
MCG       CRU4     SA5    A4+2        R1RS = R1 + 2                                 FCC1555X      10     A
MCG                BX6    X5                                                        FCC1555X      11     A
MCG                LX6    59-R1.SOP-SO.LKP                                          FCC1555X      12     A
MCG                PL     X6,CRU0     IF SOLK[R1RS] " 1  */ NOT A TEMP LOCK         FCC1555X      13     A
MCG                RJ     FBR         FIND ANOTHER B-REGISTER                       FCC1555X      14     A
MCG                MI     B3,CRU      IF ACT < 0    */ NONE FOUND                   FCC1555X      15     A
MCG                SB2    B3                                                        FCC1555X      16     A
MCG                EQ     CRU0                                                      FCC1555X      17     A
MCG       FBR      TITLE  FBR - FIND AN AVAILABLE B-REGISTER ( *RJRS* ONLY )        FCC0641       19     A
MCG       **       FBR - FIND AN AVAILABLE B-REGISTER FOR AN INSTRUCTION THAT       FCC0641       20     A
MCG       *                       PRECEEDS A *RJRS*.                                FCC0641       21     A
MCG       *                                                                         FCC0641       22     A
MCG       *        ENTRY  (A5,X5) = R1RS, [R1RS] , R1 WORD OF THE RJ *RS*            FCC0641       23     A
MCG       *                                                                         FCC0641       24     A
MCG       *        EXIT   (B3) = ACT , < 0 IF NO B-REG WAS FOUND                     FCC0641       25     A
MCG       *                (X6) = REGNO , OF THE B-REGISTER                          FCC0641       26     A
MCG                                                                                FCC0641       27     A
MCG       FBR      ROUTINE                                                           FCC0641       28     A
MCG                LX5    -R1.SOP                                                   FCC0641       29     A
MCG                MX2    -SO.REGL                                                  FCC0641       30     A
MCG                BX6    -X2*X5      REGNO = SOREG[R1RS]                           FCC0641       31     A
MCG                RJ     CBR         ACT = CBR(REGNO)                              FCC0641       32     A
MCG                PL     B3,FBR      IF ACT > 0                                    FCC0641       33     A
MCG                                                                                FCC0641       34     A
MCG                SX6    7           REGNO = 7                                     FCC0641       35     A
MCG       FBR1     RJ     CBR         ACT = CBR(REGNO)                              FCC0641       36     A
MCG                PL     B3,FBR2     IF ACT > 0                                    FCC0641       37     A
MCG                SX6    X6-1        REGNO = REGNO - 1                             FCC0641       38     A
MCG                NZ     X6,FBR1     IF REGNO " 0                                  FCC0641       39     A
MCG                EQ     FBR                                                       FCC0641       40     A
MCG                                                                                FCC0641       41     A
MCG       FBR2     BX5    X2*X5                                                     FCC0641       42     A
MCG                BX7    X6+X5       SOREG[R1RS] = REGNO                           FCC0641       43     A
MCG                LX7    R1.SOP                                                    FCC0641       44     A
MCG                SA7    A5                                                        FCC0641       45     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                PRINT  II,(* FBR, NEW REGNO =*Z2),X6                    FCC1555X      18     A
MCG                EQ     FBR                                              FCC0641       46     A
MCG         CBR    SPACE  3                                                FCC0641       47     A
MCG        **      CBR - CHECK B-REGISTER TO SEE IF IT IS AVAILABLE        FCC0641       48     A
MCG        *                                                               FCC0641       49     A
MCG        *       ENTRY  (X6) = REGNO                                     FCC0641       50     A
MCG        *                                                               FCC0641       51     A
MCG        *       EXIT   (B3) = ACT < 0 IF NOT AVAIL                      FCC0641       52     A
MCG                                                                        FCC0641       53     A
MCG         CBR    ROUTINE                                                 FCC0641       54     A
MCG                SB3    X6+         ACT = REGNO                          FCC0641       55     A
MCG                SA1    RVT+X6                                           FCC0641       56     A
MCG                ZR     X1,CBR      IF RVT(I) = 0                        FCC0641       57     A
MCG                SA1    "TB"+X1     LIP = TB + RI[RVT(I)]                FCC0641       58     A
MCG                MX3    -I.USESL                                         FCC0641       59     A
MCG                LX3    I.USESP                                          FCC0641       60     A
MCG                BX3    -X3*X1                                           FCC0641       61     A
MCG                NZ     X3,CBR1     IF USES[LIP] " 0                     FCC0641       62     A
MCG                SA1    A1-B1       DIP = LIP - 1                        FCC0641       63     A
MCG                LX1    59-D.PRSP                                        FCC0641       64     A
MCG                PL     X1,CBR1     IF ^PRS[DIP]                         FCC0641       65     A
MCG                SA1    A1+2        R1P = DIP + 2                        FCC0641       66     A
MCG                LX1    59-R1.SOP-SO.SFLP                                FCC0641       67     A
MCG                MI     X1,CBR      IF SFL[R1P]   */ CAN REUSE THE REGISTER FCC0641    68     A
MCG                                                                        FCC0641       69     A
MCG         CBR1   SB3    -B1         ACT = -1                             FCC0641       70     A
MCG                EQ     CBR                                              FCC0641       71     A
MCG         FXR    TITLE  FXR - FIND AN AVAILABLE X-REGISTER               MCG         1238     A
MCG        **      FXR - FIND A FREE X-REGISTER                            MCG         1239     A
MCG        *                                                               MCG         1240     A
MCG        *       ENTRY  (X5) = REGISTER CLASS BITS                       MCG         1241     A
MCG        *              (X4) = SHIFT( [DI], -D.TYP )                    MCG         1242     A
MCG        *              (X7) = AXR & REGS IN CLASS                      MCG         1243     A
MCG        *              (B5) = T(IS)                                    MCG         1244     A
MCG        *                                                               MCG         1245     A
MCG        *       EXIT   (X3) = 1/FOUND,NOT FOUND,41/REG BIT,18/REGISTER NUMBER MCG   1246     A
MCG        *              (B4,B5) = T(IS),T(EX) ARE UPDATED                MCG         1247     A
MCG        *                                                               MCG         1248     A
MCG        *       USES   B - 2, 3  , A - 2, 3                             MCG         1249     A
MCG        *              X - 1, 2, 3, 5, 6, 7                             MCG         1250     A
MCG                                                                        MCG         1251     A
MCG         FXR0   SB5    X1          T(IS) = BT                           MCG         1252     A
MCG                BX3    X6                                               MCG         1253     A
MCG                SB4    X1+         T(EX) = BT                           MCG         1254     A
MCG                                                                        MCG         1255     A
MCG         FXR    ROUTINE                                                 MCG         1256     A
MCG         #MD    IFGE   .CPU,74                                  ?66/76 MCG         1257     A
MCG                MI     X3,FXR2     IF REGNO < 0  */ NOTHING AVAIL IN THIS CLASS MCG 1258     A
MCG                                                                        MCG         1259     A
MCG        *       SEARCH FOR REGISTER IN CLASS WITH EARLIEST AVAILABLE TIME. MCG         1260     A
MCG                                                                        MCG         1261     A
MCG                BX6    X3                                               MCG         1262     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SX1    B2          BT = TRA(REGNO)                                  MCG        1263     A
MCG                                                                                    MCG        1264     A
MCG        FXR1    AX3    18                                                           MCG        1265     A
MCG                BX7    X7-X3       AVAIL = AVAIL - REGBIT(REGNO)                    MCG        1266     A
MCG                NO                                                                  MCG        1267     A
MCG                NX2    B2,X7                                                        MCG        1268     A
MCG                SA3    B2+FXRA+16-47   REGNO = FXRA( NORMC(AVAIL) )                 MCG        1269     A
MCG                MI     X3,FXR0                                                      MCG        1270     A
MCG                SA2    TRA+X3                                                       MCG        1271     A
MCG                SB3    X2                                                           MCG        1272     A
MCG                IX5    X2-X1                                                        MCG        1273     A
MCG                LE     B3,B5,FXR   IF TRA(REGNO) < T(IS)                           MCG        1274     A
MCG                                                                                    MCG        1275     A
MCG                PL     X5,FXR1     IF BT @ TRA(REGNO)                               MCG        1276     A
MCG                BX6    X3                                                           MCG        1277     A
MCG                SX1    B3          BT = TRA(REGNO)                                  MCG        1278     A
MCG                EQ     FXR1                                                         MCG        1279     A
MCG        #MD     ENDIF                                                '66/76        MCG        1280     A
MCG                                                                                    MCG        1281     A
MCG       *        CHECK REMAINING USES OF OPERANDS, SEE IF THEY ARE IN THE            MCG        1282     A
MCG       *        CLASS OF REGISTERS REQUESTED, AND WILL BE RELEASED BY ISSUE         MCG        1283     A
MCG       *        OF THE CURRENT INSTRUCTION ( MAY BE REUSED ).                       MCG        1284     A
MCG                                                                                    MCG        1285     A
MCG        FXR2    SA2    LXR                                                          MCG        1286     A
MCG                MX7    -D.TYL                                                       MCG        1287     A
MCG                BX6    -X2*X5      CLASS = ^LXR & CLASS                             MCG        1288     A
MCG                BX5    -X7*X4                                                       MCG        1289     A
MCG                ZR     X5,FXR3     IF TYPE[DI] = I                                  MCG        1290     A
MCG                AX7    B1,X5                                                        MCG        1291     A
MCG                ZR     X7,FXR      IF TYPE[DI] = II  */ NO OPERANDS                 MCG        1292     A
MCG        FXR     SPACE  1,10                                                         MCG        1293     A
MCG       *        CHECK TYPE III INSTRUCTION ( NOTE - RI IS A DEF )                   MCG        1294     A
MCG                                                                                    MCG        1295     A
MCG                SA5    "DI"-B1     R2 = DI - 1                                      MCG        1296     A
MCG                LX5    -IH.RFP                                                      MCG        1297     A
MCG                BX7    "RN"X5      RF = RF[R2]                                      MCG        1298     A
MCG                ZR     X7,FXR      IF RF = 0                                        MCG        1299     A
MCG                SA2    "TB"+X7     LP = BASE + RF                                   MCG        1300     A
MCG                SA1    RBV+X2      REGB = RBV( REG[LP] )                            MCG        1301     A
MCG                MX3    -I.USESL+1                                                   MCG        1302     A
MCG                BX6    X1*X6                                                        MCG        1303     A
MCG                ZR     X6,FXR      IF REGB & CLASS = 0                              MCG        1304     A
MCG                SB3    X2                                                           MCG        1305     A
MCG                LX2    -I.USESP-1                                                   MCG        1306     A
MCG                BX7    -X3*X2                                                       MCG        1307     A
MCG                NZ     X7,FXR      IF USES[LP] " 1                                  MCG        1308     A
MCG                                                                                    MCG        1309     A
MCG        FXR2A   SA2    TRA+B3                                                       MCG        1310     A
MCG                SX3    B3          REGNO = REG[LP]                                  MCG        1311     A
MCG                SB2    X2                                                           MCG        1312     A
MCG                LE     B2,B5,FXR   T(IS/EX) = MAX( T(IS/EX) , TRA(REGNO) )         MCG        1313     A
MCG                SB4    X2                                                           MCG        1314     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SB5    X2                                                  MCG        1315     A
MCG                EQ     FXR                                                 MCG        1316     A
MCG                                                                           MCG        1317     A
MCG       *        CHECK OPERANDS OF TYPE I INSTRUCTION                        MCG        1318     A
MCG                                                                           MCG        1319     A
MCG        FXR3    BX2    X4                                                  MCG        1320     A
MCG                SA5    "DI"-2       R1 = DI - 2                            MCG        1321     A
MCG                LX2    D.TYP-D.TYIOP                                       MCG        1322     A
MCG                BX7    -X7*X2                                              MCG        1323     A
MCG                SA1    TYIOM+X7                                            MCG        1324     A
MCG                MX3    I.USESL-1                                           MCG        1325     A
MCG                UX2    B2,X1                                               MCG        1326     A
MCG                BX5    X2*X5                                               MCG        1327     A
MCG                LX3    I.USESL+I.USESP                                     MCG        1328     A
MCG                AX5    B2,X5        RNUMS = TYIOM[DI] & [R1]               MCG        1329     A
MCG                SB3    0            RG = 0                                 MCG        1330     A
MCG                SB2    -T(INF)      TB = *INFINITY*                        MCG        1331     A
MCG                                                                           MCG        1332     A
MCG        FXR4    BX7    "RN"X5       R = RN[RNUMS]                          MCG        1333     A
MCG                AX5    R1.RIL       RNUMS = SHIFT(RNUMS,-16)               MCG        1334     A
MCG                ZR     X7,FXR5      IF R = 0                               MCG        1335     A
MCG                SA2    "TB"+X7      LP = BASE + R                          MCG        1336     A
MCG                BX1    X3*X2                                               MCG        1337     A
MCG                ZR     X1,FXR6      IF USES[LP] = 1                        MCG        1338     A
MCG                BX7    X5-X7                                               MCG        1339     A
MCG                ZR     X7,FXR7      IF RJ = RK                             MCG        1340     A
MCG        FXR5    NZ     X5,FXR4      IF RNUMS " 0                           MCG        1341     A
MCG                NZ     B3,FXR2A     IF RG " 0     */ WINNER FOUND          MCG        1342     A
MCG                SX3    -B1          REGNO = -1                             MCG        1343     A
MCG                EQ     FXR                                                 MCG        1344     A
MCG                                                                           MCG        1345     A
MCG        FXR6    SA1    RBV+X2                                              MCG        1346     A
MCG                BX7    X1*X6                                               MCG        1347     A
MCG                ZR     X7,FXR5      IF CLASS & RBV(REG[LP]) = 0            MCG        1348     A
MCG                SA1    TRA+X2                                              MCG        1349     A
MCG                SX7    X1+B2                                               MCG        1350     A
MCG                PL     X7,FXR5      IF TRA(REG[LP]) \ TB                   MCG        1351     A
MCG                BX1    -X1                                                 MCG        1352     A
MCG                SB3    X2           RG = REG[LP]                           MCG        1353     A
MCG                SB2    X1           TB = TRA(RG)                           MCG        1354     A
MCG                EQ     FXR5                                                MCG        1355     A
MCG                                                                           MCG        1356     A
MCG       *        TYPE I  &  RJ = RK                                          MCG        1357     A
MCG                                                                           MCG        1358     A
MCG        FXR7    SB3    X2           REGNO = REG[LP]                        MCG        1359     A
MCG                LX2    -I.USESP                                            MCG        1360     A
MCG                MX3    -I.USESL                                            MCG        1361     A
MCG                BX2    -X3*X2                                              MCG        1362     A
MCG                SX7    X2-2                                                MCG        1363     A
MCG                NZ     X7,FXR       IF USES[LP] " 2                        MCG        1364     A
MCG                                                                           MCG        1365     A
MCG                SA2    RBV+B3                                              MCG        1366     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                BX6    X2*X6                                                  MCG        1367    A
MCG                ZR     X6,FXR      IF RBV(REGNO) & CLASS = 0                  MCG        1368    A
MCG                EQ     FXR2A                                                  MCG        1369    A
MCG                TTL    MCG - MACHINE CODE GENERATOR - INSTRUCTION ISSUE       MCG        1370    A
MCG       PJI      TITLE  PJI - PROCESS JUMP INSTRUCTION                         MCG        1371    A
MCG      **        PJI - PROCESS JUMP INSTRUCTION                                MCG        1372    A
MCG      *                                                                       MCG        1373    A
MCG      *         ENTRY  (A4,X4) = DI, SHIFT( [DI] , -D.TYP )                   MCG        1374    A
MCG      *                                                                       MCG        1375    A
MCG      *         EXIT   INSTRUCTION ISSUED, *ICL* UPDATED, CLOCK ADVANCED, ETC. MCG        1376    A
MCG                                                                              MCG        1377    A
MCG       PJI      ROUTINE                                                       MCG        1378    A
MCG                MX1    -D.TYL                                                 MCG        1379    A
MCG                MX7    0                                                      MCG        1380    A
MCG                SB3    B0          REGNO = 0                                  MCG        1381    A
MCG                BX6    X1+X4                                                  MCG        1382    A
MCG                SA7    TRS         TRS = 0                                    MCG        1383    A
MCG                ZR     X6,PJI2     IF TYPE[DI] = IV                           MCG        1384    A
MCG                                                                              MCG        1385    A
MCG      *         DETERMINE EXECUTION TIME OF CONDITIONAL JP                    MCG        1386    A
MCG                                                                              MCG        1387    A
MCG                SA1    "DI"-B1     R2 = DI - 1                                MCG        1388    A
MCG                LX1    -IH.RFP                                                MCG        1389    A
MCG                BX6    "RN"X1      R = RF[R2]                                 MCG        1390    A
MCG                ZR     X6,PJI1     IF R = 0                                   MCG        1391    A
MCG                SA3    "TB"+X6     LJ = BASE + R                              MCG        1392    A
MCG                LX3    -I.REGP                                                MCG        1393    A
MCG                SA2    TRA+X3                                                 MCG        1394    A
MCG                SB2    X2                                                     MCG        1395    A
MCG                LE     B2,B4,PJI1  T(EX) = MAX( T(EX) , TRA(REG[LJ]) )        MCG        1396    A
MCG                SB4    X2                                                     MCG        1397    A
MCG       PJI1     SA1    A1-B1       R1 = R2 - 1                                MCG        1398    A
MCG                BX6    "RN"X1      R = RI[R1]                                 MCG        1399    A
MCG                SA3    "TB"+X6     LJ = BASE + R                              MCG        1400    A
MCG                LX3    -I.REGP                                                MCG        1401    A
MCG                SB3    X3          REGNO = REG[LJ]                            MCG        1402    A
MCG                SA2    TRA+X3                                                 MCG        1403    A
MCG                SB2    X2                                                     MCG        1404    A
MCG                LE     B2,B4,PJI2  T(EX) = MAX( T(EX) , TRA(REGNO) )          MCG        1405    A
MCG                SB4    X2                                                     MCG        1406    A
MCG                                                                              MCG        1407    A
MCG       PJI2     SA5    "DI"+B1     LI = DI + 1                                MCG        1408    A
MCG                RJ     AIL         ADJUST THE ISSUE LIST                      MCG        1409    A
MCG                RJ     DUC         DECREMENT USES COUNTS                      MCG        1410    A
MCG                SA5    "DI"-2      R1 = DI - 2                                MCG        1411    A
MCG                RJ     SII         SAVE THE INSTRUCTION                       MCG        1412    A
MCG                                                                              MCG        1413    A
MCG       DBG      IF     DEF,/DEBUG/II                                          MCG        1414    A
MCG                SX7    B4                                                     MCG        1415    A
MCG                PX6    B3,X7                                                  MCG        1416    A
MCG                LX7    PB.TEXP                                                MCG        1417    A
MCG                BX6    X7+X6                                                  MCG        1418    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                 SA6    PBI                                                  MCG          1419     A
MCG                 PRNT   PJI,II,SAV                                           MCG          1420     A
MCG        DBG      ENDIF                                                       MCG          1421     A
MCG                                                                             MCG          1422     A
MCG                 SA5    "DI"-2      R1 = DI - 2                              MCG          1423     A
MCG                 UX5    B2,X5                                                MCG          1424     A
MCG                 SX7    B2-OC.JIN                                            MCG          1425     A
MCG                 MI     X7,PJI5     IF OC[R1] < OC.JIN  */ A CONDITIONAL JP  MCG          1426     A
MCG        PJI      SPACE  1,10                                                 MCG          1427     A
MCG       *         PROCESS UNCONDITIONAL JUMPS ( UJP / RJX / JIN )             MCG          1428     A
MCG                                                                             MCG          1429     A
MCG                 S"DI"  "DI"                                                 CCGA006        31     A
MCG                 RJ     SSP         SET N.SP FOR UPCOMING SEQUENCE           CCGA006        32     A
MCG                 RJ     AVC         ADVANCE CLOCK                            MCG          1430     A
MCG                 SA1    JLR                                                  MCG          1431     A
MCG                 ZR     X1,PJI      IF JLR = 0    */ NO UJP LOCKED REGISTERS MCG          1432          I
           -FCC1442
MCG                 MX6    0                                                    MCG          1433          I
           -FCC1442
MCG                 SA3    AXR                                                  MCG          1434          I
           -FCC1442
MCG                 MX7    13                                                   MCG          1435          I
           -FCC1442
MCG                 SA6    /JAM/ISC    ISC = 0       */ CLEAR ISSUED ST CHAIN   FCC0625         9          I
           -FCC1442
MCG                                                                             MCG          1436          I
           -FCC1442
MCG                 SA3    AXR                                                  FCC1442         6     A
MCG                 MX6    0                                                    FCC1442         7     A
MCG                 SA6    /JAM/ISC    ISC = 0       */ CLEAR ISSUED STORE CHAIN FCC1442        8     A
MCG                 MX7    13                                                   FCC1442         9     A
MCG                 ZR     X1,PJI      IF JLR = 0    */ NO UJP LOCKED REGISTERS FCC1442        10     A
MCG                                                                             FCC1442        11     A
MCG        PJI4     PX1    X1                                                   MCG          1437     A
MCG                 NX5    B2,X1       REGNO = NORMC( JLR )                     MCG          1438     A
MCG                 SA2    RBV+B2                                               MCG          1439     A
MCG                 AX5    B2,X7                                                MCG          1440     A
MCG                 BX3    X2+X3       AXR = RBV(REGNO) ! AXR                   MCG          1441     A
MCG                 SA6    RVT+B2      RVT(REGNO) = 0                           MCG          1442     A
MCG                 BX1    -X5*X1      JLR = ^MASK(13+REGNO) & JLR              MCG          1443     A
MCG                 NZ     X1,PJI4     IF JLR " 0                               MCG          1444     A
MCG                 BX7    X3                                                   MCG          1445     A
MCG                 SA7    A3                                                   MCG          1446     A
MCG                 SA6    A1                                                   MCG          1447     A
MCG                 EQ     PJI                                                  MCG          1448     A
MCG                                                                             MCG          1449     A
MCG       *         CONDITIONAL JUMP, ADVANCE CLOCK, PARCEL COUNT               MCG          1450     A
MCG                                                                             MCG          1451     A
MCG        PJI5     SA4    "DI"                                                 MCG          1452     A
MCG                 MX1    -D.FTL                                               MCG          1453     A
MCG                 SA2    PC                                                   MCG          1454     A
MCG                 LX4    -D.FTP                                               MCG          1455     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                BX6    -X1*X4                                                  MCG        1456     A
MCG                SX7    X6+B4       T = T(EX) + FT[DI]                          MCG        1457     A
MCG                SX6    B1+B1                                                   MCG        1458     A
MCG                SX3    X2-3                                                    MCG        1459     A
MCG                SA7    T                                                       MCG        1460     A
MCG                PL     X3,PJI6     PC = IF( PC = 3) 2 ; ELSE MOD( PC+2 , 4 )   MCG        1461     A
MCG                MX3    -2                                                      MCG        1462     A
MCG                IX2    X2+X6                                                   MCG        1463     A
MCG                BX6    -X3*X2                                                  MCG        1464     A
MCG       PJI6     SA6    A2                                                      MCG        1465     A
MCG                EQ     PJI                                                     MCG        1466     A
MCG       PPI      TITLE  PPI - PROCESS PSEUDO INSTRUCTION                        MCG        1467     A
MCG      **        PPI - PROCESS PSEUDO INSTRUCTION                               MCG        1468     A
MCG      *         ENTRY  (A5,X5) = R1,[R1]                                       MCG        1469     A
MCG      *                (A4,X4) = DI, SHIFT([DI], -D.TYP )                      MCG        1470     A
MCG                                                                               MCG        1471     A
MCG                PROCESS (BOS,EOS)                                              MCG        1472     A
MCG       PPI0     SA5    "DI"+B1     LI = DI + 1                                 MCG        1473     A
MCG                RJ     AIL         CALL AIL( PI , LI , REGNO )                 MCG        1474     A
MCG                RJ     SSP         SET N.SP FOR UPCOMING SEQUENCE              CCGA006      33     A
MCG                                                                               MCG        1475     A
MCG       PPI      ROUTINE                                                        MCG        1476     A
MCG                PRNT   PPI,II,SAVPBI                                           MCG        1477     A
MCG                UX7    B2,X5                                                   MCG        1478     A
MCG                SB3    B0          REGNO = 0                                   MCG        1479     A
MCG                JP     PPI.JT+B2   JUMP( PPI.JT( OC[R1] ) )                    MCG        1480     A
MCG       DAR      SPACE  2                                                       MCG        1481     A
MCG      *         DAR  RI,RJ - DEFINE A-REGISTER R-NUMBER                        MCG        1482     A
MCG                                                                               MCG        1483     A
MCG                PROCESS DAR                                                    MCG        1484     A
MCG                SA4    "DI"                                                    MCG        1485     A
MCG                LX5    -R1.RJP                                                 MCG        1486     A
MCG                BX5    "RN"X5      RJ = RJ[R1]                                 MCG        1487     A
MCG                SA3    "TB"+X5     LJ = RJ + BASE                              MCG        1488     A
MCG                LX3    -I.REGP                                                 MCG        1489     A
MCG                SA1    ALR                                                     MCG        1490          I
          -FCC0603
MCG                SA2    RBV+X3                                                  MCG        1491          I
          -FCC0603
MCG                SB3    X3+AMX      AREGNO = REG[LJ] + AMX                      MCG        1492          I
          -FCC0603
MCG                BX6    -X2*X1      ALR = ^RBV(REG[LJ]) & ALR                   MCG        1493          I
          -FCC0603
MCG                SA6    A1                                                      MCG        1494          I
          -FCC0603
MCG                SA1    RVT+X3                                                  MCG        1495          I
          -FCC0603
MCG                SB4    B0          T(EX) = 0     */ MAKE X-REG AVAIL AT T = 0  MCG        1496          I
          -FCC0603
MCG                MX2    1                                                       MCG        1497          I
          -FCC0603
MCG                BX6    -X2*X1      LOCK[RVT(REGNO)] = 0                        MCG        1498          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
          -FCC0603
MCG                SA6    A1                                                    MCG        1499          I
          -FCC0603
MCG                SB3    X3+AMX      REGNO = REG[LJ] + AMX  */ A-REGISTER NUMBER  FCC0603     99     A
MCG                SB4    0           T(EX) = 0     */ MAKE X-REG AVAIL AT T = 0  FCC0603    100     A
MCG                RJ     DUC         DECREMENT USES COUNTS                       MCG        1500     A
MCG                SB4    B5          T(EX) = T(IS)                               MCG        1501     A
MCG                EQ     PPI0                                                    MCG        1502     A
MCG         DEF    SPACE  2                                                       MCG        1503     A
MCG        *       DEF  RI,SOREG      DEFINE RI TO BE IN *SOREG*                  MCG        1504     A
MCG                                                                               MCG        1505     A
MCG                PROCESS DEF                                                    MCG        1506     A
MCG                BX7    "RN"X5      RI = RI[R1]                                 MCG        1507     A
MCG                LX5    -R1.SOP                                                 MCG        1508     A
MCG                MX3    -SO.RTL-SO.RNL                                          MCG        1509     A
MCG                BX6    -X3*X5                                                  MCG        1510     A
MCG                SB3    X6          REGNO = SOREG[R1]                           MCG        1511     A
MCG                SA7    RVT+X6      RVT(REGNO) = RI                             MCG        1512     A
MCG                LX5    58-SO.LKP                                               MCG        1513     A
MCG                PL     X5,PPI4     IF SOLK[R1] = 0  */ NOT A FULL LOCK         MCG        1514          I
          -FCC0603
MCG                                                                               MCG        1515          I
          -FCC0603
MCG                SA1    LXR                                                     MCG        1516          I
          -FCC0603
MCG                SA2    RBV+B3                                                  MCG        1517          I
          -FCC0603
MCG                MX3    1                                                       MCG        1518          I
          -FCC0603
MCG                SA1    AXR                                                     FCC0603    101     A
MCG                SA2    RBV+B3                                                  FCC0603    102     A
MCG                BX6    -X2*X1      AXR = ^RBV(REGNO) & AXR                     FCC0603    103     A
MCG                SA6    A1                                                      FCC0603    104     A
MCG                PL     X5,PPI0     IF SOLK[R1] = 0  */ NOT FULL /A-REG LOCK    FCC0603    105     A
MCG                LX5    1                                                       FCC0603    106     A
MCG                MI     X5,PPI4     IF SOLK[R1] = 3  */ A-REG LOCK              FCC0603    107     A
MCG                                                                               FCC0603    108     A
MCG                MX3    1                                                       FCC0603    109     A
MCG                SA1    LXR                                                     FCC0603    110     A
MCG                BX7    X3+X7       LOCK[RVT(REGNO)] = 1                        MCG        1519     A
MCG                SA7    A7                                                      MCG        1520     A
MCG                BX6    X1+X2       LXR = LXR ! RBV(REGNO)                      MCG        1521     A
MCG                SA6    A1                                                      MCG        1522     A
MCG                                                                               MCG        1523          I
          -FCC0603
MCG         PPI4   SA1    AXR                                                     MCG        1524          I
          -FCC0603
MCG                SA2    RBV+B3                                                  MCG        1525          I
          -FCC0603
MCG                BX6    -X2*X1      AXR = ^RBV(REGNO) & AXR                     MCG        1526          I
          -FCC0603
MCG                SA6    A1                                                      MCG        1527          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

          -FCC0603
MCG                EQ     PPI0                                                  MCG         1528          I
          -FCC0603
MCG                EQ     PPI0                                                  FCC0603      111     A
MCG                                                                             FCC0603      112     A
MCG        PPI4    SA1    ALR                                                   FCC0603      113     A
MCG                BX6    -X2*X1      ALR = ^RBV(REGNO) & ALR                   FCC0603      114     A
MCG                SA6    A1                                                    FCC0603      115     A
MCG                EQ     PPI0                                                  FCC0603      116     A
MCG        LAB     SPACE  3,10                                                  MCG         1529     A
MCG       *        LAB/ENT  IH,CA                                               MCG         1530     A
MCG                                                                             MCG         1531     A
MCG                PROCESS (LAB,ENT)                                            MCG         1532     A
MCG                SA5    "DI"+B1     LI = DI + 1                               MCG         1533     A
MCG                RJ     AIL         CALL AIL( PI , LI , 0 )                   MCG         1534     A
MCG                RJ     SSP         SET N.SP FOR UPCOMING SEQUENCE            CCGA006       34     A
MCG                SA5    "DI"-2      R1 = DI - 2                               MCG         1535     A
MCG                RJ     SII         SAVE THE ISSUED INSTRUCTION               MCG         1536     A
MCG                IFNE   .IWT,0                                                MCG         1537     A
MCG                SA3    "DI"-B1     R2 = DI - 1                               MCG         1538     A
MCG                LX3    59-IH.STKP                                            MCG         1539     A
MCG                PL     X3,PPI6     IF ^STK[R2]   */ LOOP NOT INSTACK         MCG         1540     A
MCG                MX6    0                                                     MCG         1541     A
MCG                SA6    IWT         IWT = 0                                   MCG         1542     A
MCG        PPI6    BSS    0                                                     MCG         1543          I
          -CCGA006
MCG                ENDIF                                          '6600         MCG         1544     A
MCG                RJ     AVC         ADVANCE CLOCK                             MCG         1545          I
          -CCGA006
MCG        PPI6    RJ     AVC         ADVANCE THE CLOCK                         CCGA006       35     A
MCG                EQ     PPI                                                   MCG         1546     A
MCG        RS      SPACE  2                                                     MCG         1547     A
MCG       *        RS  RI,LK,SOREG - REGISTER STORE PROCESSING                  MCG         1548     A
MCG                                                                             MCG         1549     A
MCG                PROCESS RS                                                   MCG         1550     A
MCG                BX7    "RN"X5      RI = RI[R1]                               MCG         1551     A
MCG                LX5    -R1.SOP                                               MCG         1552     A
MCG                MX3    -SO.RTL-SO.RNL                                        MCG         1553     A
MCG                BX6    -X3*X5      REGNO = SOREG[R1]                         MCG         1554     A
MCG                LX5    -SO.LKP                                               MCG         1555     A
MCG                MX3    -SO.LKL                                               MCG         1556     A
MCG                BX2    -X3*X5                                                MCG         1557     A
MCG                SB2    X2                                                    MCG         1558     A
MCG                JP     PPI9+B2     JUMP( SOLK[R1] )                          MCG         1559          I
          -FCC0603
MCG                                                                             MCG         1560          I
          -FCC0603
MCG        PPI9    BSS    0                                                     MCG         1561          I
          -FCC0603
MCG                JP     PPI7+B2     JUMP( SOLK[R1] )                          FCC0603      117     A
MCG                                                                             FCC0603      118     A
MCG        PPI7    BSS    0                                                     FCC0603      119     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                 LOC    0                                                    MCG         1562     A
MCG                 SA1    JLR                                                  MCG         1563     A
MCG                 EQ     PPI10                                                MCG         1564     A
MCG                 EQ     PPI0        */ TEMP LOCK                             MCG         1565     A
MCG                 EQ     PPI11                                                MCG         1566     A
MCG                 LOC    *O                                                   MCG         1567     A
MCG                                                                             MCG         1568     A
MCG       *         RJ REGISTER STORE                                            MCG         1569     A
MCG                                                                             MCG         1570     A
MCG                 SA3    "DI"-3      */ CHECK USES OF R-NUMBER OF X-REG OF PRED MCG         1571     A
MCG                 LX3    -I.REGP                                              MCG         1572     A
MCG                 SA1    TRA+X3      XREG = REG[DI-3]                         MCG         1573     A
MCG                 MX2    -I.USESL                                             MCG         1574     A
MCG                 LX3    I.REGP-I.USESP                                       MCG         1575     A
MCG                 SA7    RVT+X6      RVT(REGNO) = RI                          MCG         1576     A
MCG                 SA7    /JAM/LRN+X6 LRN(REGNO) = RI                          FCC0641       72     A
MCG                 SB3    X6                                                   MCG         1577     A
MCG                 BX6    X1                                                   MCG         1578     A
MCG                 SA6    TRA+B3      TRA(REGNO) = TRA(XREG)                   MCG         1579     A
MCG                 BX6    -X2*X3                                               MCG         1580     A
MCG                 NZ     X6,PPI4     IF USES[DI-3] " 0                        MCG         1581          I
          -FCC0603
MCG                 NZ     X6,PPI8     IF USES[DI-3] " 0  */ X-RESULT USED      FCC0603      120     A
MCG                 LX3    I.USESP-I.REGP                                       MCG         1582     A
MCG                 SA1    RBV+X3                                               MCG         1583     A
MCG                 SA2    AXR                                                  MCG         1584     A
MCG                 SA6    RVT+X3      RVT( REG[DI-3] ) = 0                     MCG         1585     A
MCG                 BX7    X1+X2                                                MCG         1586     A
MCG                 SA7    A2          AXR = AXR ! RBV(XREG)                    MCG         1587     A
MCG                 EQ     PPI4                                                 MCG         1588          I
          -FCC0603
MCG       PPI8      SA1    AXR                                                  FCC0603      121     A
MCG                 SA2    RBV+B3                                               FCC0603      122     A
MCG                 BX6    -X2*X1      AXR = ^RBV(REGNO) & AXR                  FCC0603      123     A
MCG                 SA6    A1                                                   FCC0603      124     A
MCG                 EQ     PPI0                                                 FCC0603      125     A
MCG       RS        SPACE  2                                                    MCG         1589     A
MCG       *         UJP LOCK - SET TO CLEAR WHEN NEXT UJP/RJX/JIN ENCOUNTERED    MCG         1590     A
MCG                                                                             MCG         1591     A
MCG       PPI10     SB2    X6-47                                                MCG         1592     A
MCG                 SX3    B1                                                   MCG         1593     A
MCG                 AX2    B2,X3                                                MCG         1594     A
MCG                 BX6    X2+X1       JLR = SHIFT(1,47-REGNO) ! JLR            MCG         1595     A
MCG                 SA6    A1                                                   MCG         1596     A
MCG                 EQ     PPI0                                                 MCG         1597     A
MCG                                                                             MCG         1598     A
MCG       *         FULL LOCK - SET LOCK BITS IN *LXR* AND *RVT*                 MCG         1599     A
MCG                                                                             MCG         1600     A
MCG       PPI11     SA5    RVT+X6                                               MCG         1601     A
MCG                 SA1    RBV+X6                                               MCG         1602     A
MCG                 SA3    LXR                                                  MCG         1603     A
MCG                 MX2    1                                                    MCG         1604     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                BX6    X2+X5       LOCK[RVT(REGNO)] = 1                          MCG        1605     A
MCG                SA6    A5                                                        MCG        1606     A
MCG                SA5    "TB"+X7     LI = BASE + RI                                MCG        1607     A
MCG                LX2    1+I.USESP                                                 MCG        1608     A
MCG                BX7    X1+X3       LXR = LXR ! RBV(REGNO)                        MCG        1609     A
MCG                SA7    A3                                                        MCG        1610     A
MCG                IX6    X5-X2       USES[LI] = USES[LI] - 1                       MCG        1611     A
MCG                SA6    A5                                                        MCG        1612     A
MCG                EQ     PPI0                                                      MCG        1613     A
MCG        EOQ     SPACE  2                                                         MCG        1614     A
MCG       *        EOQ - TERMINATE *PIT* BUFFER, EXIT TO EOQ ADDRESS                MCG        1615     A
MCG                                                                                 MCG        1616     A
MCG                PROCESS EOQ                                                      MCG        1617     A
MCG                SA5    PPIB                                                      MCG        1618     A
MCG                RJ     SII         SII( NOP )                                    MCG        1619     A
MCG                SA3    IIP                                                       MCG        1620     A
MCG                SA4    A3+B1                                                     MCG        1621     A
MCG                IX6    X3-X4       L.PIT = IIP(1) - IIP(2)                       MCG        1622     A
MCG                SA1    EOQA                                                      MCG        1623     A
MCG                SA6    L.PIT                                                     MCG        1624     A
MCG                SB2    X1                                                        MCG        1625     A
MCG                JP     B2          JUMP( EOQA )                                  MCG        1626     A
MCG                                                                                 MCG        1627     A
MCG        PPIB    VFD    12/2000B+OC.NOP,48/0                                      MCG        1628     A
MCG        AIL     TITLE  AIL - ADJUST ISSUE LIST                                   MCG        1629     A
MCG       **       AIL - ADJUST ISSUE CANDIDATE LIST TO REMOVE ISSUED INSTRUCTION   MCG        1630     A
MCG       *        FROM IT. NOTE THAT BECAUSE THE SUCCESSORS OF AN INSTRUCTION      MCG        1631     A
MCG       *        ARE SORTED BY "LATE START TIME" IN THE USES INDEX TABLE, THE     MCG        1632     A
MCG       *        *ICL* WILL ALSO BE SORTED BY "LST".                              MCG        1633     A
MCG       *                                                                         MCG        1634     A
MCG       *        ENTRY  (A5,X5) = LI,[LI] - LINK WORD OF ISSUED INSTRUCTION.      MCG        1635     A
MCG       *               (B3) = REGNO ( NEW VALUE OF LINK FIELD )                  MCG        1636     A
MCG       *                                                                         MCG        1637     A
MCG       *        EXIT   (A4,X4) = DI, [DI] - DESCRIPTOR OF ISSUED INSTRUCTION.    MCG        1638     A
MCG       *               (X0) = R-NUMBER EXTRACTOR MASK                            MCG        1639     A
MCG       *               LINK[LI] = REGNO                                          MCG        1640     A
MCG       *               ICL = ISSUE CANDIDATE LIST POINTER UPDATED.               MCG        1641     A
MCG       *                                                                         MCG        1642     A
MCG       *        USES   B - 2, 6   , X - ALL  ( RESETS X0 )                       MCG        1643     A
MCG                                                                                 MCG        1644     A
MCG                                                                                 MCG        1645     A
MCG       *        RESET *ICL* POINTER, AND SET EXIT CONDITIONS                     MCG        1646     A
MCG                                                                                 MCG        1647     A
MCG        AIL0    MX0    -R1.RIL+1                                                MCG        1648          I
          -CCG11
MCG                SA4    A5-B1       DI = LI - 1                                   MCG        1649          I
          -CCG11
MCG                LX0    1           (X0) = R-NUMBER EXTRACTOR MASK                MCG        1650          I
          -CCG11
MCG        AIL0    SA4    A5-B1       DI = LI - 1                                   CCG11       124     A
MCG                SX0    RN.MASK     (X0) = R-NUMBER MASK                          CCG11       125     A
MCG                SA1    "TB"                                                      MCG        1651     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                NZ     B6,AIL      IF PS " 0     */ DIDNT ISSUE FIRST ELEMENT          MCG        1652     A
MCG                SX6    X1          ICL = LINK[*BOS*]                                   MCG        1653     A
MCG                IX7    X1-X6                                                           MCG        1654     A
MCG                SA6    ICL                                                             MCG        1655     A
MCG                SA7    "TB"+       LINK[*BOS*] = 0                                     MCG        1656     A
MCG                                                                                       MCG        1657     A
MCG        AIL     ROUTINE                                                                MCG        1658     A
MCG                SA1    "PI"+"TB"                                                       MCG        1659     A
MCG                SB6    "PI"        PS = PI                                             MCG        1660     A
MCG                MX7    -I.LINKL                                                        MCG        1661     A
MCG                S"PI"  "PI"+"TB"   PI = PI + BASE                                      MCG        1662     A
MCG                SX3    X5                                                              MCG        1663     A
MCG                BX2    X7*X1                                                           MCG        1664     A
MCG                MX4    1                                                               MCG        1665     A
MCG                IX6    X2+X3       LINK[PI] = LINK[LI]                                 MCG        1666     A
MCG                LX4    I.IIP+1     II[LI] = 1    */ INDICATE ISSUED                    MCG        1667     A
MCG                SA3    "TB"+X3     NI = LINK[LI] + BASE                                MCG        1668     A
MCG                BX5    X7*X5                                                           MCG        1669     A
MCG                SA6    A1                                                              MCG        1670     A
MCG                SX2    B3                                                              MCG        1671     A
MCG                LX2    I.REGP                                                          MCG        1672     A
MCG                IX4    X2+X4       REG[LI] = REGNO                                     MCG        1673     A
MCG                SX0    B1                                                              MCG        1674     A
MCG                BX6    X4+X5                                                           MCG        1675     A
MCG                SA6    A5                                                              MCG        1676     A
MCG                MX7    -I.INDXL                                                        MCG        1677     A
MCG                SA2    O.TREE                                                          MCG        1678     A
MCG                LX5    -I.INDXP                                                        MCG        1679     A
MCG                BX1    -X7*X5                                                          MCG        1680     A
MCG                LX7    I.LSTP                                                          MCG        1681     A
MCG                IX2    X1+X2                                                           MCG        1682     A
MCG                BX4    -X7*X3      RN = LST[NI]                                        MCG        1683     A
MCG                SA2    X2          UI = O.TREE + INDX[LS]                              MCG        1684          I
          -FCC1056
MCG                SA2    X2          UI = O.TREE + INDX[LI]                              FCC1056      33     A
MCG                EQ     AIL3                                                            MCG        1685     A
MCG                                                                                       MCG        1686     A
MCG        *       LINK THIS TO NEXT, LAST TO THIS                                        MCG        1687     A
MCG                                                                                       MCG        1688     A
MCG         AIL1   SX5    A3-"TB"                                                         MCG        1689     A
MCG                LX0    I.ICP                                                           MCG        1690     A
MCG                IX5    X0+X5       IC[SI] = 1                                          MCG        1691     A
MCG                BX6    X5+X6       LINK[SI] = NI - BASE                                MCG        1692     A
MCG                MX5    -I.LINKL                                                        MCG        1693     A
MCG                SA6    A1                                                              MCG        1694     A
MCG                LX0    -I.ICP                                                          MCG        1695     A
MCG                SA1    "PI"        LINK[PI] = SI - BASE                                MCG        1696     A
MCG                S"PI"  A6          PI = SI                                             MCG        1697     A
MCG                BX6    X5*X1                                                           MCG        1698     A
MCG                SX5    A6-"TB"                                                         MCG        1699     A
MCG                IX6    X6+X5                                                           MCG        1700     A
MCG                                                                                       MCG        1701     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG      *         IF NOT LAST SUCCESSOR, ADVANCE TO NEXT                    MCG      1702     A
MCG                                                                          MCG      1703     A
MCG       AIL2     SA6    A1                                                 MCG      1704     A
MCG                MI     X2,AIL0     IF LAST[UI]                            MCG      1705     A
MCG                SA2    A2+B1       UI = UI + 1                            MCG      1706     A
MCG                                                                          MCG      1707     A
MCG      *         REDUCE PRED COUNT OF SUCCESSORS OF INSTRUCTION.          MCG      1708     A
MCG                                                                          MCG      1709     A
MCG       AIL3     SA1    "TB"+X2     SI = SUCC[UI] + BASE                   MCG      1710     A
MCG                IX6    X1-X0       NPRED[SI] = NPRED[SI] - 1              MCG      1711     A
MCG                SB2    X1+                                                MCG      1712     A
MCG                NE     B2,B1,AIL2  IF NPRED[SI] " 0                       MCG      1713     A
MCG                                                                          MCG      1714     A
MCG      *         ADD SUCCESSOR TO LIST BEFORE NEXT WITH GREATER *LST*     MCG      1715     A
MCG                                                                          MCG      1716     A
MCG                BX1    -X7*X1                                             MCG      1717     A
MCG                IX5    X1-X4                                              MCG      1718     A
MCG       AIL4     MI     X5,AIL1     IF LST[SI] < RN                        MCG      1719     A
MCG                S"PI"  A3          PI = NI                                MCG      1720     A
MCG                SA3    "TB"+X3     NI = LINK[NI] + BASE                   MCG      1721     A
MCG                BX4    -X7*X3      RN = LST[NI]                           MCG      1722     A
MCG                IX5    X1-X4                                              MCG      1723     A
MCG                EQ     AIL4                                               MCG      1724     A
MCG       DUC      TITLE  DUC - DECREMENT USES COUNTS                        MCG      1725     A
MCG      **        DUC - DECREMENT USES COUNTS                              MCG      1726     A
MCG      *                                                                   MCG      1727     A
MCG      *         ENTRY  (A4,X4) = DI, [DI]                                MCG      1728     A
MCG      *                (B3) = REGNO  ( OF RESULT REGISTER )              MCG      1729     A
MCG      *                (B4) = T(EX) OF INSTRUCTION                       MCG      1730     A
MCG      *                                                                   MCG      1731     A
MCG      *         EXIT   RVT, AXR UPDATED                                  MCG      1732     A
MCG      *                                                                   MCG      1733     A
MCG      *         USES   B - 2, 6   X - 1 - 7                              MCG      1734     A
MCG                                                                          MCG      1735     A
MCG       DUC0     MI     X4,DUC      IF ^D[DI] ! TYPE[DI] = IV              MCG      1736     A
MCG                SA5    "DI"-2      R1 = DI - 2                            MCG      1737     A
MCG                SA1    RBV+B3                                             MCG      1738     A
MCG                SA2    AXR                                                MCG      1739     A
MCG                BX7    "RN"X5                                             MCG      1740     A
MCG                SA7    B6+B3       RVT(REGNO) = RI                        MCG      1741     A
MCG                BX6    -X1*X2      AXR = ^RBV(REGNO) & AXR                MCG      1742     A
MCG                SA6    A2                                                 MCG      1743     A
MCG                                                                          MCG      1744     A
MCG       DUC      ROUTINE                                                   MCG      1745     A
MCG                LX4    58-D.TYP                                           MCG      1746     A
MCG                IX7    X4+X4                                              MCG      1747     A
MCG                SB6    RVT                                                MCG      1748     A
MCG                MX6    -D.TYIOL                                           MCG      1749     A
MCG                MI     X7,DUC0     IF TYPE[DI] = II ! IV                  MCG      1750     A
MCG                                                                          MCG      1751     A
MCG                SA5    "DI"-2      R1 = DI - 2                            MCG      1752     A
MCG                PL     X4,DUC1     IF TYPE[DI] = I                        MCG      1753     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                                                                    MCG     1754   A
MCG     *         TYPE III INSTRUCTION                                 MCG     1755   A
MCG                                                                    MCG     1756   A
MCG               SA3    "DI"-B1     R2 = DI - 1                       MCG     1757   A
MCG               LX4    1+D.TYP-D.^DP                                 MCG     1758   A
MCG               BX7    "RN"X5      RI = RI[R1]                       MCG     1759   A
MCG               LX3    -IH.RFP                                       MCG     1760   A
MCG               BX5    "RN"X3      RNUMS = RF[R2]                    MCG     1761   A
MCG               PL     X4,DUC2     IF DEF[DI]                        MCG     1762   A
MCG               LX7    R1.RIL                                        MCG     1763   A
MCG               BX5    X7+X5       RNUMS = SHIFT(RI,16) ! RNUMS      MCG     1764   A
MCG               EQ     DUC2                                          MCG     1765   A
MCG                                                                    MCG     1766   A
MCG     *         TYPE I INSTRUCTION                                   MCG     1767   A
MCG                                                                    MCG     1768   A
MCG      DUC1     LX4    2+D.TYP-D.TYIOP                               MCG     1769   A
MCG               BX7    -X6*X4                                        MCG     1770   A
MCG               SA3    TYIOM+X7                                      MCG     1771   A
MCG               UX2    B2,X3                                         MCG     1772   A
MCG               LX4    D.TYIOP+59-D.^DP                              MCG     1773   A
MCG               BX1    X2*X5                                         MCG     1774   A
MCG               AX5    B2,X1       RNUMS = TYIOM[DI] & [R1]          MCG     1775   A
MCG                                                                    MCG     1776   A
MCG     *         SETUP REGISTERS FOR LOOP                             MCG     1777   A
MCG                                                                    MCG     1778   A
MCG      DUC2     MX1    I.USESL-1                                     MCG     1779   A
MCG               SX7    B1                                            MCG     1780   A
MCG               LX1    I.USESL+I.USESP                               MCG     1781   A
MCG               BX3    "RN"X5      R = RI[RNUMS]                     MCG     1782   A
MCG               LX7    I.USESP                                       MCG     1783   A
MCG               NZ     X5,DUC4     IF RNUMS " 0                      MCG     1784   A
MCG               EQ     DUC0                                          MCG     1785   A
MCG                                                                    MCG     1786   A
MCG     *         DECREMENT USES COUNTS OF OPERANDS, ADJUST *AXR*, *RVT* MCG     1787   A
MCG                                                                    MCG     1788   A
MCG      DUC3     AX5    R1.RIL      RNUMS = SHIFT(RNUMS,-R1.RIL)      MCG     1789   A
MCG               ZR     X5,DUC0     IF RNUMS = 0                      MCG     1790   A
MCG               BX3    "RN"X5      R = RI[RNUMS]                     MCG     1791   A
MCG      DUC4     SA2    "TB"+X3     LP = BASE + R                     MCG     1792   A
MCG               ZR     X3,DUC3     IF R = 0                          MCG     1793   A
MCG               IX6    X2-X7       USES[LP] = USES[LP] - 1           MCG     1794   A
MCG               BX3    X1*X2                                         MCG     1795   A
MCG               SA6    A2                                            MCG     1796   A
MCG               NZ     X3,DUC3     IF USES[LP] " 0                   MCG     1797   A
MCG               LX2    -I.REGP     REGP = REG[LP]                    MCG     1798   A
MCG               SA3    B6+X2                                         MCG     1799   A
MCG               MI     X3,DUC3     IF LOCK[RVT(REGP)]                MCG     1800   A
MCG                                                                    MCG     1801   A
MCG     *         RETURN REGISTER TO AVAILABLE LIST                    MCG     1802   A
MCG                                                                    MCG     1803   A
MCG               MX6    0                                             MCG     1804   A
MCG               SA6    A3          RVT(REGP) = 0                     MCG     1805   A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SX6    B4+B1                                                  MCG        1806     A
MCG                SA6    TRA+X2      TRA(REGNO) = T(EX) + 1                     MCG        1807     A
MCG                SA3    RBV+X2                                                 MCG        1808     A
MCG                SA2    AXR                                                    MCG        1809     A
MCG                BX6    X3+X2                                                  MCG        1810     A
MCG                SA6    A2          AXR = RBV(REGP) ! AXR                      MCG        1811     A
MCG                EQ     DUC3                                                   MCG        1812     A
MCG       ASC      TITLE  ASC - ADVANCE SIMULATION CLOCKS                        MCG        1813     A
MCG      **        ASC - ADVANCE SIMULATION CLOCKS FOR ISSUED INSTRUCTION        MCG        1814     A
MCG      *         ALSO DO LD/ST TO SLD/SST TRANSFORMATION AND ADJUST *SP* COUNT MCG        1815     A
MCG      *         ENTRY  (A4,X4) = DI, [DI]                                     MCG        1816     A
MCG      *                (B4) = T(EX)                                           MCG        1817     A
MCG      *                (B5) = T(IS)                                           MCG        1818     A
MCG      *                (B3) = REGNO ( SOREGNO OF *RI* REGISTER )              MCG        1819     A
MCG      *                                                                       MCG        1820     A
MCG      *         EXIT   (B3) = CHANGED TO A-REG NUMBER FOR LD/ST INSTRUCTIONS  MCG        1821     A
MCG      *                T, PC, TRA, TFU, LST, EXT. UPDATED.                    MCG        1822     A
MCG                                                                             MCG        1823     A
MCG       ASC0     SA6    A1                                                     MCG        1824     A
MCG                IX7    X3+X7                                                  MCG        1825     A
MCG                PRNT   ASC,II                                                 MCG        1826     A
MCG                PRNT   ASC,ICL                                                MCG        1827     A
MCG                PRNT   ASC,RUI                                                MCG        1828     A
MCG                SA7    T                                                      MCG        1829     A
MCG                                                                             MCG        1830     A
MCG       ASC      ROUTINE                                                       MCG        1831     A
MCG                MX7    -D.LDSTL                                               MCG        1832     A
MCG                LX4    -D.LDSTP                                               MCG        1833     A
MCG                SA2    PBI+1                                                  MCG        1834     A
MCG                BX1    -X7*X4                                                 MCG        1835     A
MCG                ZR     X1,ASC7     IF LDST[DI] = 0                            MCG        1836     A
MCG       #DAL     IFNE   .DAL,0                                                 MCG        1837     A
MCG                BX3    X4                                                     MCG        1838     A
MCG                LX3    D.LDSTP+59-D.L2P                                       MCG        1839     A
MCG                MI     X3,ASC6     IF L2[DI]                                  MCG        1840     A
MCG       #DAL     ENDIF                                                         MCG        1841     A
MCG                                                                             MCG        1842     A
MCG      *         COMMON HANDLING OF LD/ST INSTRUCTIONS                         MCG        1843     A
MCG                                                                             MCG        1844     A
MCG                MX7    -SO.RNL                                                MCG        1845     A
MCG                SX3    B3                                                     MCG        1846     A
MCG                AX1    D.LDP-D.LDSTP     I = LD[DI]                           MCG        1847     A
MCG                UX5    B2,X2       F = FU[PBI+1]                              MCG        1848     A
MCG                BX3    -X7*X3      RN = SORN[REGNO]                           MCG        1849     A
MCG                SX7    B4+"INCT"   TFU(F) = T(EX) + INC TIME                  MCG        1850     A
MCG                LX4    D.LDSTP-D.SZP                                          MCG        1851     A
MCG                SA2    TSL+X1                                                 MCG        1852     A
MCG                SB3    X3+RN.A0    REGNO = A0 + RN                            MCG        1853     A
MCG       #MD      IFGE   .CPU,74                                       ?66/76  MCG        1854     A
MCG                SA7    TFU+B2                                                 MCG        1855     A
MCG                SA7    TRA+RN.A0+X3      TRA(A0+RN) = T(EX) + INC TIME         MCG        1856     A
MCG                SX6    X2+B4                                                  MCG        1857     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SA6    TRA+RN.X0+X3      TRA(X0+RN) = T(EX) + TSL(I)                MCG        1858     A
MCG                SA6    LST+X1      LST(I) = T(EX) + TSL(I)                          MCG        1859     A
MCG       #MD      ENDIF                                                '66/76        MCG        1860     A
MCG                                                                                   MCG        1861     A
MCG       #MD      IFEQ   .CPU,76                                       ?7600         MCG        1862     A
MCG                SA2    T                                                            MCG        1863     A
MCG                SA3    TRS         BIT = 0                                          MCG        1864     A
MCG                SX5    B1                        */ ADJUST BIT VECTOR BASE          MCG        1865     A
MCG                IX7    X5-X2                                                        MCG        1866     A
MCG                SB6    B4+X7                                                        MCG        1867     A
MCG                AX6    B6,X3       TRS = SHIFT( TRS , -( T(EX)+1-T ) )              MCG        1868     A
MCG                SA6    A3                                                           MCG        1869     A
MCG       #MD      ENDIF                                                '7600         MCG        1870     A
MCG                                                                                   MCG        1871     A
MCG      *         CHECK FOR TRANSFORMATION TO SHORT LD/ST                              MCG        1872     A
MCG                                                                                   MCG        1873     A
MCG                MX5    D.ARL                                                        MCG        1874     A
MCG                LX5    D.ARL+D.ARP-D.SZP                                            MCG        1875     A
MCG                SA3    "DI"-B1     R2 = DI - 1                                      MCG        1876     A
MCG                BX1    X5*X4                                                        MCG        1877     A
MCG                LX6    X3                                                           MCG        1878          I
          -FCC0881
MCG                MX7    1                                                            FCC0881      30     A
MCG                BX6    -X7*X3      LD[R2W] = 0   */ CLEAR APL LD BIT                FCC0881      31     A
MCG                ZR     X1,ASC5     IF ^(SR[DI] ! FP[DI])                            MCG        1879     A
MCG                SB2    ARA-1                                                        MCG        1880     A
MCG                SA6    B2+B1       ARA(0) = [R2] */ SET SEARCH TERMINATOR           MCG        1881     A
MCG                PL     X1,ASC3     IF ^SR[DI]                                       MCG        1882     A
MCG                                                                                   MCG        1883     A
MCG                SA2    ARA+7       I = ARA(7)                                       MCG        1884     A
MCG       ASC1     BX5    X6-X2                                                        MCG        1885     A
MCG                SA2    A2-B1       I = I - 1                                        MCG        1886     A
MCG                NZ     X5,ASC1     IF [I] " [R2]                                    MCG        1887     A
MCG                SX7    A2-B2                                                        MCG        1888     A
MCG                ZR     X7,ASC2     IF I = 0      */ MATCH WITH SEARCH TERMIN        MCG        1889     A
MCG                SX4    B1          SZ[DI] = 1                                       MCG        1890     A
MCG                SA7    RMR         RMR = I - ARA(-1) */ SAVE A REG NUM              MCG        1891     A
MCG                EQ     ASC5                                                         MCG        1892     A
MCG                                                                                   MCG        1893     A
MCG      *         CHECK FOR ELIMINATION OF F.P. ADDSUB                                 MCG        1894     A
MCG                                                                                   MCG        1895     A
MCG       ASC2     LX1    D.SRP-D.FPP                                                  MCG        1896     A
MCG                PL     X1,ASC5     IF ^FP[DI]                                       MCG        1897     A
MCG       ASC3     SA2    ARA+7                                                        MCG        1898     A
MCG                MX7    -IH.CAL                                                      MCG        1899     A
MCG                LX7    IH.CAP                                                       MCG        1900     A
MCG       ASC4     BX5    X6-X2                                                        MCG        1901     A
MCG                SA2    A2-1        I = I - 1                                        MCG        1902     A
MCG                BX3    X7*X5                                                        MCG        1903     A
MCG                NZ     X3,ASC4     IF RFIH[I] " RFIH[R2]                            MCG        1904     A
MCG                                                                                   MCG        1905     A
MCG                SX1    A2-B2       I = I - ARA(-1)                                  MCG        1906     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SA2    A2+B1                                                    MCG          1907     A
MCG                ZR     X1,ASC5     IF I = 0                                     MCG          1908     A
MCG                SA6    ARA+B3-RN.A0                                             MCG          1909     A
MCG                LX6    -IH.CAP                                                  MCG          1910     A
MCG                SB6    X6                                                       MCG          1911     A
MCG                LX2    -IH.CAP                                                  MCG          1912     A
MCG                SB2    X2                                                       MCG          1913     A
MCG                SX6    B6-B2       CA[R2] = CA[R2] - CA[I]                      MCG          1914     A
MCG                LX6    IH.CAP                                                   MCG          1915     A
MCG                BX7    -X7*X6                                                   MCG          1916     A
MCG                IX6    X7+X1                                                    MCG          1917     A
MCG                SA6    RMR         RMR = IHINFO(0,CA,AREGNO)                    MCG          1918     A
MCG                EQ     ASC9                                                     MCG          1919     A
MCG                                                                               MCG          1920     A
MCG        ASC5    SA6    ARA+B3-RN.A0                                             MCG          1921     A
MCG                EQ     ASC9                                                     MCG          1922     A
MCG                                                                               MCG          1923     A
MCG      *         LEVEL 2 LD/ST ( DRL/DWL )                                       MCG          1924     A
MCG                                                                               MCG          1925     A
MCG        #DAL    IFNE   .DAL,0                                                   MCG          1926     A
MCG        ASC6    SA1    L2T+X1-1                                                 MCG          1927          I
          -FCCA546X
MCG        ASC6    BSS    0
>>>>     FCCA546X13                                                               FCCA546X        7     A
MCG      #MD       IFGT   .CPU,74
>>>>     FCCA546X14                                                               FCCA546X        8     A
MCG                SA1    L2T+X1-1
>>>>     FCCA546X15                                                               FCCA546X        9     A
MCG                BX6    X1                                                       MCG          1928     A
MCG                SA6    FUT+9       FUT(LCM) = L2T(LDST)  */ SET LD/ST TIME      MCG          1929     A
MCG      #MD       ENDIF
>>>>     FCCA546X17                                                               FCCA546X       10     A
MCG        #DAL    ENDIF                                                           MCG          1930     A
MCG        ASC     SPACE  3                                                        MCG          1931     A
MCG      *         GENERAL INSTRUCTION                                             MCG          1932     A
MCG                                                                               MCG          1933     A
MCG        ASC7    LX4    D.LDSTP+59-D.PSP                                         MCG          1934     A
MCG                UX2    B2,X2       F = FU[PBI+1]                                MCG          1935     A
MCG                PL     X4,ASC8     IF ^PS[DI]                                   MCG          1936     A
MCG                SA3    N.SP                                                     MCG          1937     A
MCG                SX6    X3+B1       N.SP = N.SP + 1                              MCG          1938     A
MCG                SA6    A3                                                       MCG          1939     A
MCG                NZ     X6,ASC8     IF N.SP " 0                                  MCG          1940     A
MCG                SA6    STL         STL = 0       */ CLEAR STORE LOCKOUT FLAG    MCG          1941     A
MCG                                                                               MCG          1942     A
MCG        ASC8    LX4    1+D.PSP-D.FTP                                            MCG          1943          I
          -CCG3
MCG        ASC8    LX4    D.PSP-D.ECJP                                             CCG3          139     A
MCG                MI     X4,ASC10    IF ECJ[DI]                                   CCG3          140     A
MCG                LX4    1+D.ECJP-D.FTP                                           CCG3          141     A
MCG                MX5    -D.FTL                                                   MCG          1944     A
MCG                BX3    -X5*X4                                                   MCG          1945     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SX6    B4+X3       TRA(REGNO) = T(EX) + FT[DI]                      MCG        1946    A
MCG                LX4    D.FTP-D.SZP                                                  MCG        1947    A
MCG        #MD     IFLE   .CPU,74                                      ?6600          MCG        1948    A
MCG                SX7    X6+B1       TFU(F) = T(EX) + FT[DI] + 1   */ 6600            MCG        1949    A
MCG        #MD     ELSE                                                '66?76         MCG        1950    A
MCG                SA2    T                                                            MCG        1951    A
MCG                SA1    TRS                                                          MCG        1952    A
MCG                SX5    B1                                                           MCG        1953    A
MCG                IX3    X6-X2                                                        MCG        1954    A
MCG                IX7    X5-X2                                                        MCG        1955    A
MCG                SB6    X3                                                           MCG        1956    A
MCG                LX5    B6,X5                                                        MCG        1957    A
MCG                SB6    B4+X7                                                        MCG        1958    A
MCG                BX1    X5+X1       TRS = TRS ! SHIFT( 1 , TRA(REGNO) - T )          MCG        1959    A
MCG                AX7    B6,X1       TRS = SHIFT( TRS , -( T(EX)+1 - T ) )            MCG        1960    A
MCG                SA2    FUT+B2-1                                                     MCG        1961    A
MCG                SA7    A1                                                           MCG        1962    A
MCG                SX7    B4+X2       TFU(F) = T(EX) + FUT(F)       */ 7600            MCG        1963    A
MCG        #MD     ENDIF                                               '7600          MCG        1964    A
MCG        #MD     IFGE   .CPU,74                                      ?66/76         MCG        1965    A
MCG                SA6    TRA+B3                                                       MCG        1966    A
MCG                SA7    TFU+B2                                                       MCG        1967    A
MCG        #MD     ENDIF                                                               MCG        1968    A
MCG                                                                                    MCG        1969    A
MCG       *        ADVANCE SIMULATION CLOCK, AND PARCEL COUNT                          MCG        1970    A
MCG                                                                                    MCG        1971    A
MCG        ASC9    SA1    PC                                                           MCG        1972    A
MCG                MX5    -2                                                           MCG        1973    A
MCG                BX2    -X5*X4      NP = SZ[DI]                                      MCG        1974    A
MCG                IX6    X1+X2       PC = PC + NP                                     MCG        1975    A
MCG        #MD     IFEQ   .CPU,76                                      ?7600          MCG        1976    A
MCG                SX7    B5+1        T = T(EX) + 1                                    MCG        1977    A
MCG        #MD     ELSE                                                ?6600          MCG        1978    A
MCG                SX7    B5+X2       T = T(IS) + NP                                   MCG        1979    A
MCG        #MD     ENDIF                                               '6600          MCG        1980    A
MCG                BX3    X5*X6                                                        MCG        1981    A
MCG                ZR     X3,ASC0     IF PC < 4                                        MCG        1982    A
MCG                SA3    IWT                                                          MCG        1983    A
MCG                BX6    -X5*X6      PC = MOD( PC , 4 )                               MCG        1984    A
MCG                ZR     X6,ASC0     IF PC = 0                                        MCG        1985    A
MCG                BX6    X2          PC = NP                                          MCG        1986    A
MCG                MX3    0           */ INTERWORD DELAY TAKEN CARE OF IN *SNI*        MCG        1987    A
MCG                EQ     ASC0                                                         MCG        1988    A
MCG                                                                                    CCG3        142    A
MCG       *        RJXJ - ASSUME A JUMP TO *+1                                         CCG3        143    A
MCG                                                                                    CCG3        144    A
MCG        ASC10   BSS    0                                                            CCG3        145    A
MCG        #MD     IFLT   .CPU,75                                                      CCG3        146    A
MCG                SX7    B4+9        T = T(EX) + 9                                    CCG3        147    A
MCG        #MD     ELSE                                                                CCG3        148    A
MCG                SA1    T                                                            CCG3        149    A
MCG                SA2    TRS                                                          CCG3        150    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SX7    B4+3        T = T(EX) + 3                                  CCG3          151     A
MCG                IX5    X7-X1                                                      CCG3          152     A
MCG                SB6    X5                                                         CCG3          153     A
MCG                SA6    B6,X2       TRS = SHIFT( TRS , -(T(NEW)-T(OLD)) )          CCG3          154         I
          -CCG11
MCG                LX6    B6,X2       TRS = SHIFT( TRS , -(T(NEW)-T(OLD)) )          CCG11         126         I
          -CCGA066
MCG                LX6    -B6,X2      TRS = SHIFT( TRS , -(T(NEW)-T(OLD)) )          CCGA066         5     A
MCG                SA6    A2                                                         CCG3          155     A
MCG       #MD      ENDIF                                                              CCG3          156     A
MCG                                                                                   CCG3          157     A
MCG                SA1    PC                                                         CCG3          158     A
MCG                MX6    0           PC = 0                                         CCG3          159     A
MCG                MX3    0                                                          CCG3          160     A
MCG                EQ     ASC0                                                       CCG3          161     A
MCG                                                                                   MCG          1989     A
MCG       #MD      IFEQ   .CPU,76                                       ?7600        MCG          1990         I
          -FCC1451
MCG       #MD      IFGT   .CPU,74                                       ?7600        FCC1451         7     A
MCG       TSL      CON    2,8         STORE/LOAD TIMES  *7600*                        MCG          1991     A
MCG                                                                                   MCG          1992     A
MCG       FUT      BSS    0           FUNCTION UNIT TIME TABLE                        MCG          1993     A
MCG                LOC    1                                                          MCG          1994     A
MCG                DATA   1,1,1,1,1                                                  MCG          1995     A
MCG                CON    2           MULTIPLY                                       MCG          1996     A
MCG                CON    1           NORM                                           MCG          1997     A
MCG                CON    18          DIVIDE                                         MCG          1998     A
MCG                DATA   1,3,1                                                      MCG          1999     A
MCG                LOC    *O                                                         MCG          2000     A
MCG                                                                                   MCG          2001     A
MCG       L2T      DATA   3,15        LEVEL 2 STORE/LOAD TIMES                        MCG          2002     A
MCG                                                                                   MCG          2003     A
MCG       #MD      ELSE                                                 '7600        MCG          2004     A
MCG       TSL      CON    9,8         STORE/LOAD TIMES  *6600*                        MCG          2005     A
MCG       #MD      ENDIF                                                '6600        MCG          2006     A
MCG       AVC      TITLE  AVC - ADVANCE CLOCK FOR UJP/RJX/LAB INSTRUCTIONS            MCG          2007     A
MCG      **        AVC - ADVANCE CLOCK AFTER ISSUE OF LAB/UJP/RJX                     MCG          2008     A
MCG      *         ENTRY  (B4) = T(EX)                                                MCG          2009     A
MCG      *         EXIT   T , PC UPDATED.  N.SP , STL SET IF STORES IN NEXT BLOCK.    MCG          2010     A
MCG                                                                                   MCG          2011     A
MCG      *         SET STORE LOCKOUT FLAG IF MORE THAN 1 STORE PRED IN NEXT BLOCK     MCG          2012     A
MCG                                                                                   MCG          2013     A
MCG       AVC0     SA4    "DI"                                                       MCG          2014     A
MCG                LX4    -D.USESP                                                   MCG          2015     A
MCG                MX5    -D.USESL                                                   MCG          2016     A
MCG                SA3    L.OTI                                                      MCG          2017     A
MCG                NZ     X3,AVC1     IF L.OTI " 0  */ JAM MODE                      MCG          2018     A
MCG                BX6    -X4+X5                                                     MCG          2019     A
MCG                AX3    B1,X6                                                      MCG          2020     A
MCG                ZR     X3,AVC1     IF N.SP @ 1                                    MCG          2021     A
MCG                SX6    X6+B1                                                      MCG          2022     A
MCG                SA6    N.SP        N.SP = -(USES[DI]-1)                           MCG          2023     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SX7    -FX.X6                                          MCG          2024     A
MCG                SA7    STL          STL = -FX.X6                       MCG          2025     A
MCG                                                                       MCG          2026     A
MCG        AVC1    SA3    "DI"-4       DL = DI - 4                        MCG          2027     A
MCG                SA1    AVCA                                            MCG          2028     A
MCG                BX2    -X1+X3                                          MCG          2029     A
MCG                ZR     X2,AVC       IF BM[DL] & TYPE[DL] = IV          MCG          2030     A
MCG                SETZERO ARA+1,7     FOR I = 1 TO 7; ARA(I) = 0         MCG          2031     A
MCG                                                                       MCG          2032     A
MCG        AVC     ROUTINE                                               MCG          2033     A
MCG                SA1    ICL                                             MCG          2034     A
MCG                MX2    -I.LSTL                                         MCG          2035     A
MCG                SA3    "TB"+X1      L = BASE + ICL                     MCG          2036     A
MCG                MX7    0                                               MCG          2037     A
MCG                SA7    PC           PC = 0                             MCG          2038     A
MCG                LX3    -I.LSTP                                         MCG          2039     A
MCG                BX6    -X2*X3                                          MCG          2040     A
MCG                SA6    T            T = LST[L]                         MCG          2041     A
MCG                SA7    TRS          TRS = 0                            MCG          2042     A
MCG                SA2    A3-3         R1 = L - 3                         MCG          2043     A
MCG                UX3    B2,X2                                           MCG          2044     A
MCG                ZR     B2,AVC       IF OC[R1] = OC.EOQ                 MCG          2045     A
MCG                SB2    X6-10                                           MCG          2046     A
MCG                GT     B2,B4,AVC0   IF T \ T(EX) + 10                  MCG          2047     A
MCG                MX2    8-1                                             MCG          2048     A
MCG                SA7    TRA+23                                          MCG          2049     A
MCG                MX6    0                                               FCC1442        12     A
MCG      +         LX2    1            FOR I = 16 TO 24, TRA(I) = 0       MCG          2050     A
MCG                SA7    A7-B1                                           MCG          2051     A
MCG                MI     X2,*                                            MCG          2052          I
          -FCC1442
MCG                SA6    A7-22+7      TRA(I-8) = 0  */ CLEAR B-REG ASSOCIATES  FCC1442        13     A
MCG                MI     X2,*-1                                          FCC1442        14     A
MCG                MX2    14-1                                            MCG          2053     A
MCG                SA7    TFU                                             MCG          2054     A
MCG      +         LX2    1            FOR I = 1 TO 14, TFU(I) = 0        MCG          2055     A
MCG                SA7    A7+B1                                           MCG          2056     A
MCG                MI     X2,*                                            MCG          2057     A
MCG                EQ     AVC0                                            MCG          2058     A
MCG                                                                       MCG          2059     A
MCG       AVCA     BFMW   D,(TY,BM)                                       MCG          2060     A
MCG       SII      TITLE  SII - SAVE ISSUED INSTRUCTION                   MCG          2061     A
MCG      **        SII - SAVE ISSUED INSTRUCTION FOR LATER PROCESSING     MCG          2062     A
MCG      *                I.E. FINAL CONVERSION OF INSTRUCTION TO MACHINE CODE.  MCG  2063     A
MCG      *                                                               MCG          2064     A
MCG      *         ENTRY  (B3) = REG[RI]                                  MCG          2065     A
MCG      *                (A5,X5) = R1, [R1]                              MCG          2066     A
MCG      *                                                               MCG          2067     A
MCG      *         EXIT   INSTRUCTION SAVED IN *PIT*, IIP UPDATED.        MCG          2068     A
MCG      *                                                               MCG          2069     A
MCG      *         USES   B - 2, 6  ,  X - 1, 2, 3, 4, 5, 6, 7            MCG          2070     A
MCG                                                                       MCG          2071     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                 QUAL   SII                                                    MCG        2072     A
MCG       SI.       SPACE  3                                                      MCG        2073          I
         -CCG11
MCG      **         SI. - SAVED INSTRUCTION FORMAT ( *PIT* )                      MCG        2074          I
         -CCG11
MCG      *                                                                        MCG        2075          I
         -CCG11
MCG      *          TYPE I     OC,0,RK,RJ,RI                                     MCG        2076          I
         -CCG11
MCG      *                                                                        MCG        2077          I
         -CCG11
MCG      *          TYPE II    OC,IN,0,0,RI                                      MCG        2078          I
         -CCG11
MCG      *                                                                        MCG        2079          I
         -CCG11
MCG      *          TYPE III   OC,CA,IH,RF,RI + *H2* IN SECOND WORD IF *H2* BIT SET MCG      2080          I
         -CCG11
MCG      *                                                                        MCG        2081          I
         -CCG11
MCG      *          TYPE IV    OC,CA,0,IH    ( SAME AS RLIST FORMAT )             MCG        2082          I
         -CCG11
MCG                                                                               MCG        2083          I
         -CCG11
MCG                 LIST   -R                                                     MCG        2084          I
         -CCG11
MCG                                                                               MCG        2085          I
         -CCG11
MCG                 DESCRIBE SI.,60    SI(OC,CA,IH/RK,RJ,RI)                      MCG        2086          I
         -CCG11
MCG       H2        DEFINE 1           *H2* FIELD IN NEXT WORD                    MCG        2087          I
         -CCG11
MCG       OC        DEFINE 11          PACKED OPCODE                              MCG        2088          I
         -CCG11
MCG       CA        DEFINE 18          CONSTANT ADDEND                            MCG        2089          I
         -CCG11
MCG       IH        DEFINE 18          SYMBOL ORDINAL                             MCG        2090          I
         -CCG11
MCG       RJ        DEFINE 6                                                      MCG        2091          I
         -CCG11
MCG       RI        DEFINE 6                                                      MCG        2092          I
         -CCG11
MCG                                                                               MCG        2093          I
         -CCG11
MCG       RK        DEQU   IH,6                                                   MCG        2094          I
         -CCG11
MCG                                                                               MCG        2095          I
         -CCG11
MCG                 LIST   R                                                      MCG        2096          I
         -CCG11
MCG       SII       EJECT                                                         MCG        2097          I
         -CCG11
MCG       SII       SPACE  3                                                      CCG11       127     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                PROCESS SA                                                  CCG2          38     A
MCG                S"DI"  A5+2        DI = R1 + 2                              CCG2          39     A
MCG                SX1    RN.A0                                                CCG2          40           I
          -CCGA073
MCG                SA1    REG=A0                                               CCGA073       11     A
MCG                LX4    59-D.TYP                                             CCG2          41     A
MCG                PL     X4,TYI      IF TYPE[DI] " II  */ NOT  *SA  RI,A0*    CCG2          42     A
MCG                LX1    SI.RJP                                               CCG2          43     A
MCG                BX7    X1+X7       REG[RJ] = RN.A0                          CCG2          44     A
MCG                EQ     SII0                                                 CCG2          45     A
MCG                                                                            CCG2          46     A
MCG       *        TYPE 1 INSTRUCTION WITH 2 OPERANDS                          MCG         2098     A
MCG                                                                            MCG         2099     A
MCG        TYI     LX5    -R1.RJP                                              MCG         2100     A
MCG                BX6    "RN"X5                                               MCG         2101     A
MCG                SA1    "TB"+X6                                              MCG         2102     A
MCG                LX1    -I.REGP                                              MCG         2103     A
MCG                LX5    R1.RJP-R1.RKP                                        MCG         2104     A
MCG                BX4    "RN"X5                                               MCG         2105     A
MCG                SA2    "TB"+X4                                              MCG         2106     A
MCG                LX2    -I.REGP                                              MCG         2107     A
MCG                SX6    X1                                                   MCG         2108     A
MCG                SX4    X2                                                   MCG         2109     A
MCG                LX6    SI.RJP                                               MCG         2110     A
MCG                BX1    X6+X7       [IIP-1] = SI(OC,0,REG[RK],REG[RJ],REG[RI])  MCG        2111     A
MCG                LX4    SI.RKP                                               MCG         2112     A
MCG                IX7    X1+X4                                                MCG         2113     A
MCG                                                                            MCG         2114     A
MCG        SII0    SA7    X3+                                                  MCG         2115     A
MCG                                                                            MCG         2116     A
MCG        SII     ROUTINE            ** ENTRY/EXIT **                         MCG         2117     A
MCG                SA3    IIP                                                  MCG         2118     A
MCG                UX4    B2,X5                                                MCG         2119     A
MCG                SX2    B3                                                   MCG         2120     A
MCG                SA1    SIIA+B2                                              MCG         2121     A
MCG                SX6    X3+B1       IIP = IIP + 1                            MCG         2122     A
MCG                PX7    B2,X2       SIW = PACK(OC[R1],0,0,REG[R])            MCG         2123     A
MCG                SB6    X1                                                   MCG         2124     A
MCG                SA6    A3                                                   MCG         2125     A
MCG                JP     B6          JUMP( SIIA( OC[R1] ) )                   MCG         2126     A
MCG        TYPEI   SPACE  3                                                    MCG         2127           I
          -CCG11
MCG        SII     SPACE  3,14                                                 CCG11        128     A
MCG       *        TYPE 1 - RK = 0                                             MCG         2128     A
MCG                                                                            MCG         2129     A
MCG                PROCESS (XMT,XMTC,CX,DRL,DWL)                               MCG         2130           I
          -CCG16
MCG                PROCESS (DRL,DWL)                                           CCG16        252     A
MCG                SA4    A5+2        DI = R1 + 2                              CCG16        253     A
MCG                LX4    59-D.L2P                                             CCG16        254     A
MCG                MI     X4,XMT      IF L2[DI]     */ ^ A *SUB0*              CCG16        255     A
MCG                SA4    A5+B1       R2 = R1 + 1                              CCG16        256           I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

          -CCGA034
MCG                SX2    X4                                                    CCG16        257          I
          -CCGA034
MCG                SA2    A5+B1       R2 = R1 + 1
>>>>      CCGA034 11                                                            CCGA034        5     A
MCG                SX2    X2
>>>>      CCGA034 12                                                            CCGA034        6     A
MCG                LX2    SI.IHP                                                CCG16        258     A
MCG                BX7    X2+X7       IH[SIW] = IH[R2]                          CCG16        259     A
MCG                                                                             CCG16        260     A
MCG        XMT     PROCESS (XMT,XMTC,CX)                                       CCG16        261          I
          -CCGA093
MCG        XMT     PROCESS (XMT,XMTC)                                          CCGA093        7     A
MCG                LX5    -R1.RJP                                               MCG         2131     A
MCG                BX6    "RN"X5                                                MCG         2132     A
MCG                SA1    "TB"+X6                                               MCG         2133     A
MCG                LX1    -I.REGP                                               MCG         2134     A
MCG                SX2    X1                                                    MCG         2135     A
MCG                LX2    SI.RJP                                                MCG         2136     A
MCG                BX7    X2+X7       [IIP-1] = SI(OC,0,REG[RJ],REG[RI])        MCG         2137     A
MCG                EQ     SII0                                                  MCG         2138     A
MCG                                                                             CCGA093        8     A
MCG       *        TYPE 1 - SWITCH RJ TO RK                                    CCGA093        9     A
MCG                                                                             CCGA093       10     A
MCG                PROCESS (CX)                                                CCGA093       11     A
MCG                LX5    -R1.RJP                                               CCGA093       12     A
MCG                BX6    "RN"X5                                                CCGA093       13     A
MCG                SA1    "TB"+X6                                               CCGA093       14     A
MCG                LX1    -I.REGP                                               CCGA093       15     A
MCG                SX2    X1                                                    CCGA093       16     A
MCG                LX2    SI.RKP                                                CCGA093       17     A
MCG                BX7    X2+X7                                                 CCGA093       18     A
MCG                EQ     SII0                                                  CCGA093       19     A
MCG                                                                             MCG         2139     A
MCG       *        TYPE I - RJ IS A DEF                                        MCG         2140     A
MCG                                                                             MCG         2141     A
MCG                PROCESS (UP,NR,RNZ)                                         MCG         2142     A
MCG                LX5    -R1.RKP                                               MCG         2143     A
MCG                BX6    "RN"X5                                                MCG         2144     A
MCG                SA1    "TB"+X6                                               MCG         2145     A
MCG                LX5    R1.RKP-R1.RJP                                         MCG         2146     A
MCG                LX1    -I.REGP                                               MCG         2147     A
MCG                SX2    X1                                                    MCG         2148     A
MCG                BX6    "RN"X5                                                MCG         2149     A
MCG                LX2    SI.RKP                                                MCG         2150     A
MCG                IX7    X2+X7       [IIP-1] = SI(OC,REG[RK],0,REG[RI])        MCG         2151     A
MCG                ZR     X6,SII0     IF RJ[R1] = 0                             MCG         2152     A
MCG                SA1    A5+4        R1S = R1 + 4                              MCG         2153     A
MCG                MX5    -SO.RTL-SO.RNL                                        MCG         2154     A
MCG                LX1    -R1.SOP                                               MCG         2155     A
MCG                BX2    -X5*X1      REG[RJ] = SOREG[R1S]                      MCG         2156     A
MCG                LX2    SI.RJP                                                MCG         2157     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                IX7    X2+X7                                                  MCG        2158     A
MCG                EQ     SII0                                                   MCG        2159     A
MCG                                                                              MCG        2160     A
MCG       *        TYPE II INSTRUCTION                                            MCG        2161     A
MCG                                                                              MCG        2162     A
MCG        TYII    AX4    R1.INP                                                 MCG        2163     A
MCG                LX4    SI.CAP                                                 MCG        2164     A
MCG                BX7    X4+X7       [IIP-1] = SI(OC,IN[R1],0,0,REG[RI])        MCG        2165     A
MCG                EQ     SII0                                                   MCG        2166     A
MCG                                                                              MCG        2167     A
MCG                PROCESS (KLS,KRS)                                              MCG        2168     A
MCG                SA1    A5+B1       R2 = R1 + 1                                MCG        2169     A
MCG                LX1    -IH.CAP                                                MCG        2170     A
MCG                SX2    X1                                                     MCG        2171     A
MCG                LX2    SI.CAP                                                 MCG        2172     A
MCG                BX7    X2+X7       [IIP-1] = SI(OC,CA[R2],0,0,REG[RI])        MCG        2173     A
MCG                EQ     SII0                                                   MCG        2174     A
MCG                                                                              MCG        2175     A
MCG       *        TYPE III JUMP INSTRUCTIONS                                     MCG        2176     A
MCG                                                                              MCG        2177     A
MCG        TYIII   SA5    A5+B1       R2 = R1 + 1                                MCG        2178     A
MCG                MX6    -IH.CAIHL                                              MCG        2179     A
MCG                LX5    -IH.CAIHP                                              MCG        2180     A
MCG                BX4    -X6*X5                                                 MCG        2181     A
MCG                LX5    IH.CAIHP-IH.RFP                                        MCG        2182     A
MCG                SA1    "TB"+X5                                                MCG        2183     A
MCG                LX1    -I.REGP                                                MCG        2184     A
MCG                LX4    SI.IHP                                                 MCG        2185     A
MCG                SX2    X1                                                     MCG        2186     A
MCG                LX2    SI.RJP                                                 MCG        2187     A
MCG                BX6    X2+X4                                                  MCG        2188     A
MCG                IX7    X6+X7       [IIP-1] = SI(OC,0,IH[R2],REG[RF],REG[RI])  MCG        2189     A
MCG                EQ     SII0                                                   MCG        2190     A
MCG                                                                              MCG        2191     A
MCG       *        TYPE IV INSTRUCTIONS AND *NOP* ( BUFFER TERMINATOR )           MCG        2192     A
MCG                                                                              MCG        2193     A
MCG                PROCESS NOP                                                    MCG        2194     A
MCG        TYIV    BX7    X5          [IIP-1] = [R1]                             MCG        2195     A
MCG                EQ     SII0                                                   MCG        2196     A
MCG        SII     EJECT                                                          MCG        2197     A
MCG       *        MEMORY REFERENCES                                              MCG        2198     A
MCG                                                                              MCG        2199     A
MCG                PROCESS (LD,ST,STT,TLD,TST,ILD)                                MCG        2200          I
          -CCG16
MCG                PROCESS (LD,ST,STT,TLD,TST,ILD,LDC,LDV)                        CCG16       262     A
MCG                SA1    RMR                                                    MCG        2201     A
MCG                MX6    -IH.CAIHL                                              MCG        2202     A
MCG                SA2    A5+B1       R2 = R1 + 1                                MCG        2203     A
MCG                NZ     X1,LSS2     IF RMR " 0    */ REDUCABLE MEMREF          MCG        2204     A
MCG                LX2    -IH.CAIHP                                              MCG        2205     A
MCG                BX6    -X6*X2                                                 MCG        2206     A
MCG                LX2    IH.CAIHP-IH.RFP                                        MCG        2207     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SA1    "TB"+X2                                          MCG          2208     A
MCG                SB6    X2                                               MCG          2209     A
MCG                MX4    -R1.H2L                                          MCG          2210     A
MCG                LX1    -I.REGP                                          MCG          2211     A
MCG                SX2    X1                                               MCG          2212     A
MCG                NE     B6,B1,LSS1  IF RF[R2] " 1                       MCG          2213     A
MCG                SX2    RN.A0       REG[RF] = A0                         MCG          2214          I
          -CCGA073
MCG                SA2    REG=A0      REG[RF] = REG=A0                     CCGA073        12     A
MCG       LSS1     LX6    SI.IHP                                           MCG          2215     A
MCG                BX1    X6+X7                                            MCG          2216     A
MCG                LX2    SI.RJP                                           MCG          2217     A
MCG                IX7    X1+X2       [IIP-1] = SI(OC,CA,IH,REG[RF],REG[RI]) MCG        2218     A
MCG                LX5    -R1.H2P                                          MCG          2219     A
MCG                BX6    -X4*X5                                           MCG          2220     A
MCG                ZR     X6,SII0     IF H2[R1] = 0                        MCG          2221     A
MCG                                                                        MCG          2222     A
MCG                MX5    12                                               MCG          2223     A
MCG                SA6    X3+B1       [IIP] = H2[R1]                       MCG          2224     A
MCG                BX7    X5-X7       H2[IIP-1] = 1                        MCG          2225     A
MCG                SA7    X3                                               MCG          2226     A
MCG                SX6    A6+B1       IIP = IIP + 1                        MCG          2227     A
MCG                SA6    A3                                               MCG          2228     A
MCG                EQ     SII                                              MCG          2229     A
MCG                                                                        MCG          2230     A
MCG       *        REDUCABLE MEM REF                                       MCG          2231     A
MCG                                                                        MCG          2232     A
MCG       LSS2     SX2    X1                                               MCG          2233     A
MCG                MX6    0                                                MCG          2234     A
MCG                IX1    X1-X2                                            MCG          2235     A
MCG                SA6    A1          RMR = 0                              MCG          2236     A
MCG                SX4    X2+RN.A0                                         MCG          2237     A
MCG                LX1    SI.CAP-IH.CAP                                    MCG          2238     A
MCG                BX5    X1+X7                                            MCG          2239     A
MCG                LX4    SI.RJP                                           MCG          2240     A
MCG                IX7    X4+X5       [IIP-1] = SI(OC,CA[RMR],0,REG[RF],REG[RI]) MCG    2241     A
MCG                EQ     SII0                                             MCG          2242     A
MCG       PLD      SPACE  2,8                                              MCG          2243     A
MCG                PROCESS (PLD,PST)                                       MCG          2244     A
MCG                SA2    REG=A0                                           CCGA073        13     A
MCG                SX6    X2          REGJ = REG=A0                        CCGA073        14     A
MCG                SA2    A5+B1       R2 = R1 + 1                          MCG          2245     A
MCG                LX4    -R1.RJP                                          MCG          2246     A
MCG                SB6    X4                                               MCG          2247     A
MCG                BX4    "RN"X4                                           MCG          2248     A
MCG                SA1    "TB"+X4     LJ = TB + RJ[R1]                     MCG          2249     A
MCG                LX2    -IH.CAP     C = CA[R2]                           MCG          2250     A
MCG                SX6    RN.A0       REGJ = A0                            MCG          2251          I
          -CCGA073
MCG                EQ     B6,B1,PLD1  IF RJ[R1] = 1                        MCG          2252     A
MCG                LX1    -I.REGP                                          MCG          2253     A
MCG                SX6    X1+         REGJ = REG[LJ]                       MCG          2254     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                                                                                  MCG        2255    A
MCG       PLD1     SA1    A1-B1       DJ = LJ - 1                                    MCG        2256    A
MCG                LX6    SI.RJP                                                     MCG        2257    A
MCG                SB6    A1-B1       R2P = DJ - 1                                   MCG        2258    A
MCG                BX7    X6+X7                                                      MCG        2259    A
MCG                LX1    59-D.PIP                                                   MCG        2260    A
MCG                MI     X1,PLD2     IF PI[DJ]     */ PRED IS A *DEF*               MCG        2261    A
MCG                LX1    D.PIP-D.PRSP                                               MCG        2262    A
MCG                PL     X1,PLD3     IF ^PRS[DI]   */ NOT REDEF OF REG              MCG        2263    A
MCG                SB6    B6+4        R2P = R2P + 4 */ ADVANCE TO *RS*               MCG        2264    A
MCG                                                                                  MCG        2265    A
MCG      *         OPT=2 PLD/PST, BASE CA IS IN R2 WORD OF *DEF/RS*                  MCG        2266    A
MCG                                                                                  MCG        2267    A
MCG       PLD2     SA1    B6                                                         MCG        2268    A
MCG                LX1    -IH.CAP                                                    MCG        2269    A
MCG                SB2    X1                                                         MCG        2270    A
MCG                SX2    X2+B2       C = CA[R2] + ( -CA[R2P] )                      MCG        2271    A
MCG                                                                                  MCG        2272    A
MCG       PLD3     MX5    -IH.CAL                                                    MCG        2273    A
MCG                BX2    -X5*X2                                                     MCG        2274    A
MCG                LX2    SI.CAP                                                     MCG        2275    A
MCG                BX7    X2+X7       [IIP-1] = SI(OC,C,0,REGJ,REG[RI])              MCG        2276    A
MCG                EQ     SII0                                                       MCG        2277    A
MCG                                                                                  MCG        2278    A
MCG      *         CLR - GENERATE A MXI 0  OR A  BXI XI-XI                           MCG        2279    A
MCG                                                                                  MCG        2280    A
MCG                PROCESS CLR                                                       MCG        2281    A
MCG                SA1    PBI+1                                                      MCG        2282    A
MCG                SB2    OC.FMA                                                     MCG        2283    A
MCG                UX4    B6,X1                                                      MCG        2284    A
MCG                SX6    B6-BOOL                                                    MCG        2285    A
MCG                PX7    B2,X7       [IIP-1] = SI(FMA,0,0,0,0,REG[RI])              MCG        2286    A
MCG                NZ     X6,SII0     IF FU[PBI+1] " BOOL                            MCG        2287    A
MCG                SB2    OC.XOR                                                     MCG        2288    A
MCG                SX4    B3          R = REG[RI]                                    MCG        2289    A
MCG                LX4    SI.RJP                                                     MCG        2290    A
MCG                BX7    X4+X7                                                      MCG        2291    A
MCG                LX4    SI.RKP-SI.RJP                                              MCG        2292    A
MCG                BX5    X4+X7                                                      MCG        2293    A
MCG                PX7    B2,X5       [IIP-1] = SI(XOR,0,R,R,R)                      MCG        2294    A
MCG                EQ     SII0                                                       MCG        2295    A
MCG                                                                                  MCG        2296    A
MCG      *         SXT -  - GENERATE AN *XMT* OR A NOMINAL LEFT SHIFT                MCG        2297    A
MCG                                                                                  MCG        2298    A
MCG                PROCESS SXT                                                       MCG        2299    A
MCG                SA1    PBI+1                                                      MCG        2300    A
MCG                SB2    OC.XMT      OC = OC.XMT                                    MCG        2301    A
MCG                UX6    B6,X1                                                      MCG        2302    A
MCG                SX4    B6-BOOL                                                    MCG        2303    A
MCG                PX7    B2,X7                                                      MCG        2304    A
MCG                ZR     X4,.XMT     IF FU[PBI+1] = BOOL                            MCG        2305    A
MCG                LX5    -R1.RJP                                                    MCG        2306    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                BX6    "RN"X5                                          MCG        2307    A
MCG                SA1    "TB"+X6                                         MCG        2308    A
MCG                LX1    -I.REGP                                         MCG        2309    A
MCG                SB2    OC.ILS      OC = OC.ILS                         MCG        2310    A
MCG                PX7    B2,X7                                           MCG        2311    A
MCG                SX2    X1                                              MCG        2312    A
MCG                LX2    SI.RKP                                          MCG        2313    A
MCG                BX7    X2+X7       [IIP-1] = SI(OC,REG[RJ],0,REG[RI])  MCG        2314    A
MCG                EQ     SII0                                            MCG        2315    A
MCG        JUMPT   SPACE  3,14                                            MCG        2316    A
MCG       *        OPR - DEFINE MACRO TO FROM JUMP TABLES FOR *PPI* AND *SII*  MCG   2317    A
MCG                                                                       MCG        2318    A
MCG                MACRO  OPR,NAM,SS                                      MCG        2319    A
MCG                NOREF  .NAM                                            MCG        2320    A
MCG        TYP     MICRO  3,3,$SS$                                        MCG        2321    A
MCG        X       IFLT   OC.NAM,OC.XMT                                   MCG        2322    A
MCG                EQ     //.NAM            NAM                           MCG        2323    A
MCG       -        JP     TY"TYP"                                         MCG        2324    A
MCG        X       ELSE                                                   MCG        2325    A
MCG                IF     DEF,.NAM,2                                      MCG        2326    A
MCG                CON    .NAM              NAM                           MCG        2327    A
MCG                SKIP   1                                               MCG        2328    A
MCG                CON    TY"TYP"     NAM                                 MCG        2329    A
MCG        X       ENDIF                                                  MCG        2330    A
MCG                ENDM                                                   MCG        2331    A
MCG        SII     SPACE  2                                               MCG        2332    A
MCG        SIIA    BSS    0                                               MCG        2333    A
MCG                LOC    0                                               MCG        2334    A
MCG                LIST   G                                               MCG        2335    A
MCG       *CALL    OPRDEFS                                                MCG        2336    A
MCG                LIST   -G                                              MCG        2337    A
MCG                LOC    *O                                              MCG        2338    A
MCG                                                                       MCG        2339    A
MCG                QUAL                                                   MCG        2340    A
MCG                                                                       MCG        2341    A
MCG        PPI.JT  EQU    /SII/SIIA                                       MCG        2342    A
MCG        SII     EQU    /SII/SII                                        MCG        2343    A
MCG        JAM     TTL    MCG - MACHINE CODE GENERATOR - JAM MODE         MCG        2344    A
MCG        JAM     TITLE  JAM MODE / DOCUMENTION                          MCG        2345    A
MCG       **       JAM MODE - OVERVIEW/STRATEGY                           MCG        2346    A
MCG       *                                                               MCG        2347    A
MCG       *        THE GOAL OF THE JAM ALGORITHM IS TO CODE INSTRUCTION SEQUENCE  MCG  2348    A
MCG       *        IN THE FINITE NUMBER OF REGISTERS ( 8 X-REGISTERS ) THAT WE  MCG   2349    A
MCG       *        HAVE ON THE 6X/7X THOUSAND SERIES MACHINES.  THE INITIAL  MCG      2350    A
MCG       *        SEQUENCE OF INSTRUCTIONS MAY REQUIRE MORE THAN 8 REGISTERS  MCG   2351    A
MCG       *        TO CODE BECAUSE REDUNDANT OPERATION ELIMINATION HAS EXTENDED  MCG  2352    A
MCG       *        THE RANGE OF THE USES OF INSTRUCTIONS, OR MAY REQUIRE LESS,  MCG   2353    A
MCG       *        BUT THE SCHEDULER MAY HAVE PICKED AN ORDER FOR THE INSTRUCTION  MCG 2354    A
MCG       *        THAT BROUGHT ABOUT A DEADLOCK.                         MCG        2355    A
MCG       *        IN JAM MODE, SCHEDULING TAKES SECOND PRIORITY TO RESOURCE  MCG     2356    A
MCG       *        ALLOCATION.  THE INPUT TO JAM IS THE SEQUENCE OF INSTRUCTIONS  MCG 2357    A
MCG       *        LST[I] = (*TXT* INDEX[I])/2  ( INITIALLY ).            MCG        2358    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG      *            THE *SIT* IS ASSUMED TO HAVE NO DUPLICATE ENTRIES                MCG      2359     A
MCG      *         ( FOR *RIL* ) , AND OPERATIONS  WHICH PRECEED AN *RJX* REG          MCG      2360     A
MCG      *         STORES ARE LINKED TO THE RJX.  MOREOVER THE REGW FIELD OF           MCG      2361     A
MCG      *         THE DESCRIPTOR CONTAINS A COUNT OF THE NUMBER OF X-REGISTERS        MCG      2362     A
MCG      *         IN USE AT ANY GIVEN POINT.                                          MCG      2363     A
MCG      *         THE BASIC STRATEGY OF JAM MODE ASSUMES THAT THE ORDER OF THE        MCG      2364     A
MCG      *         INSTRUCTIONS IS A REASONABLE ONE AND ATTEMPTS TO ISSUE THE          MCG      2365     A
MCG      *         INSTRUCTIONS IN THAT ORDER.                                         MCG      2366     A
MCG      *         SINCE MOST PROBLEMS FROM REGISTER JAMMING ARISE FROM                MCG      2367     A
MCG      *         ATTEMPTING TO KEEP TOO MANY PARTIAL RESULTS IN THE REGISTERS        MCG      2368     A
MCG      *         FOR TOO LONG, WE ADOPT THE FOLLOWING STRATEGY.                      MCG      2369     A
MCG      *         ON ENTRY TO JAM THE INSTRUCTIONS HAVE BEEN TOTALLY ORDERED,         MCG      2370     A
MCG      *         A SIT BUILT, AND THE NUMBER OF REGISTERS IN USE AT EACH POINT       MCG      2371     A
MCG      *         COMPUTED.                                                           MCG      2372     A
MCG      *         THE ALGORITHM THEN PROCEEDS TO ISSUE THE INSTRUCTIONS IN            MCG      2373     A
MCG      *         THE GIVEN ORDER AND ADJUSTS IT AS FOLLOWS:                          MCG      2374     A
MCG      *         WHEN A RESULT, R IN A REGISTER IS THE OPERAND OF AN                 MCG      2375     A
MCG      *         INSTRUCTION WHICH DOES NOT EXHAUST ITS USES, LOOK AT THE            MCG      2376     A
MCG      *         DISTANCE AND REGISTER REQUIREMENTS BETWEEN THE ISSUED               MCG      2377     A
MCG      *         INSTRUCTION AND THE NEXT USE OF R.  IF THE USE IS                   MCG      2378     A
MCG      *         "FAR AWAY" AND TOO MANY OTHER RESULTS ARE BEING EXTENDED            MCG      2379     A
MCG      *         OVER THAT INTERVAL, THEN SAVE THE RESULT AND REISSUE THE            MCG      2380     A
MCG      *         INSTRUCTION.  IN THIS MANNER MOST CRISIS SITUATIONS ARE             MCG      2381     A
MCG      *         AVIODED AND THE SEQUENCE IS UNSQUEEZED IN A GRACEFUL FASHION.       MCG      2382     A
MCG       RV.      SPACE  3                                                            MCG      2383     A
MCG                QUAL   JAM                                                          MCG      2384     A
MCG                                                                                    MCG      2385     A
MCG                IF     DEF,/DEBUG/JAM,1                                             MCG      2386     A
MCG                TRACER (SSR,UJR)                                                    MCG      2387     A
MCG       N.UR0    EQU    9           NUMBER OF UNLOADED REGISTERS ( INIT VALUE )      MCG      2388     A
MCG                                                                                    MCG      2389     A
MCG      **        RV. - FORMAT OF *RVT* ENTRIES DURING JAM MODE                       MCG      2390     A
MCG                                                                                    MCG      2391     A
MCG                DESCRIBE RV.,60                                                     MCG      2392     A
MCG       LOCK     DEFINE 1           LOCK BIT                                         MCG      2393     A
MCG       COST     DEFINE 11          COST OF RESTORING RESULT IN REG                  MCG      2394     A
MCG                DEFINE 11                                                           MCG      2395     A
MCG       SRO      DEFINE 1           STORE REGISTER OUT ( UNSAVED RESULT )            MCG      2396     A
MCG       NU       DEFINE 18          *TXT* INDEX OF NEXT USE                          MCG      2397     A
MCG       RN       DEFINE 18          R-NUMBER IN THE REG                              MCG      2398     A
MCG       JAM      TITLE  JAM MODE / CONTROL                                           MCG      2399     A
MCG      **        JAM - REGISTER JAM CODE GENERATION MODE                             MCG      2400     A
MCG                                                                                    MCG      2401     A
MCG       JAM0     SA1    M.OT                                                         MCG      2402     A
MCG                SA2    =XN.OT                                                       MCG      2403          I
          -CCG8
MCG                MX7    0                                                            MCG      2404          I
          -CCG8
MCG                IX3    X2-X1                                                        MCG      2405          I
          -CCG8
MCG                SA7    L.OTI       L.OTI = 0                                        MCG      2406          I
          -CCG8

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                BX6    X1                                                  MCG        2407          I
          -CCG8
MCG                PL     X3,JAM          N.OT = MAX( N.OT , M.OT )           MCG        2408          I
          -CCG8
MCG                SX7    0                                                   CCG8        155     A
MCG                SA7    L.OTI       L.OTI = 0                               CCG8        156     A
MCG       .PLI     IFNE   HC.ID,3     ^ PL/I                                  CCG8        157     A
MCG                SA2    =XN.OT                                              CCG8        158          I
          -CCG18
MCG                SA2    =XN$OT                                              CCG18       174     A
MCG       .PLI     ELSE                                                       CCG8        159     A
MCG                SA4    =XCO.PN                                             CCG8        160          I
          -CCGNC
MCG                SA4    =XHO$PN                                             CCGNC       785     A
MCG                SA3    =XO.ADT                                             CCG8        161          I
          -CCGNC
MCG                SA3    =XO$ADT                                             CCGNC       786     A
MCG                IX7    X3+X4                                               CCG8        162     A
MCG                SA2    X7                                                  CCG8        163     A
MCG                LX1    18                                                  CCG8        164     A
MCG                SX3    X2                                                  CCG8        165     A
MCG                BX1    X1+X3                                               CCG8        166     A
MCG       .PLI     ENDIF                                                      CCG8        167     A
MCG                MX6    X1+X2       N.OT = MAX( N.OT , M.OT )               CCG8        168     A
MCG                SA6    A2                                                  MCG        2409     A
MCG                                                                           MCG        2410     A
MCG       JAM      ROUTINE                                                    MCG        2411     A
MCG                SETZERO M.OT,10    SETZERO( M.OT , 10 )                    MCG        2412     A
MCG                                                                           MCG        2413     A
MCG                SA4    L.TXT                                               MCG        2414     A
MCG                SA2    O.TXT                                               MCG        2415     A
MCG                SB2    X4-4                                                MCG        2416     A
MCG                SA6    X2+3        FOR I = 3 TO L.TXT BY 4; TXT(I) = 0     MCG        2417     A
MCG       +        SB2    B2-4                                                MCG        2418     A
MCG                SA6    A6+4                                                MCG        2419     A
MCG                GT     B2,*-1                                              MCG        2420     A
MCG                                                                           MCG        2421     A
MCG       .PLI     IFEQ   HC.ID,3     IF PL/I                                 CCGA006      36     A
MCG                MX6    0                                                   CCGA006      37     A
MCG       .PLI     ELSE                                                       CCGA006      38     A
MCG                SA1    =XCO.OPT                                            MCGF1         1          I
          -CCGNC
MCG                SA1    =XHO$OPT                                            CCGNC       787     A
MCG                SX6    B1                                                  MCGF1         2     A
MCG                AX1    58                -1 OR 0                           MCGF1         3     A
MCG                IX6    X6+X1       MODE = IF( CO.OPT < 2 ) THEN 1 ELSE 0   MCGF1         4          I
          -CCGNC
MCG                IX6    X6+X1       MODE = IF( HO$OPT < 2 ) THEN 1 ELSE 0   CCGNC       788     A
MCG       .PLI     ENDIF                                                      CCGA006      39     A
MCG                CALL   RIO         RESET INSTRUCTION ORDER                 MCG        2422          I
          -CCGNC
MCG                CALL   RIO#        RESET THE INSTRUCTION ORDER             CCGNC       789     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SX6    B1                                                    MCG        2423     A
MCG                CALL   BDT         FORM THE DEPENDENCY INFO                  MCG        2424          I
          -CCGNC
MCG                CALL   BDT#        FORM THE DEPENDENCY INFO                  CCGNC       790     A
MCG                SX7    JAM0                                                  MCG        2425     A
MCG                SX0    B1                                                    MCG        2426     A
MCG                RJ     PRS         PRESET STORAGE                            MCG        2427     A
MCG                SX7    N.UR0                                                 MCG        2428     A
MCG                SA7    UR          UR = N.UR0                                MCG        2429     A
MCG                MX6    0                                                     MCG        2430     A
MCG                SA6    N.SP        N.SP = 0                                  MCG        2431     A
MCG                SX7    -FX.X6                                                MCG        2432     A
MCG                SA7    STL         STL = -FX.X6                              MCG        2433     A
MCG                RJ     SMB                                                   MCG        2434     A
MCG                                                                             MCG        2435     A
MCG       *        MAIN LOOP                                                     MCG        2436     A
MCG                                                                             MCG        2437     A
MCG        JAM1    SA1    SF                                                    MCG        2438     A
MCG                SA2    AXR                                                   MCG        2439     A
MCG                BX6    X1*X2                                                 MCG        2440     A
MCG                ZR     X6,JAM2     IF SF & AXR = 0 */ NO UNSAVED STORES/ST REGS MCG        2441     A
MCG                RJ     SUR         STORE UNSAVED RESULTS                     MCG        2442     A
MCG                                                                             MCG        2443     A
MCG        JAM2    RJ     SNI         SELECT NEXT INSTRUCTION TO ISSUE          MCG        2444     A
MCG                MI     B3,JAM6     IF REGNO < 0  */ NO ISSUABLE INST FOUND   MCG        2445     A
MCG                ZR     B3,JAM4     IF REGNO = 0  */ A ISSUED PSEUDO INST     MCG        2446     A
MCG                LX5    59-I.IIP                                              MCG        2447     A
MCG                PL     X5,JAM3     IF ^II[L]                                 MCG        2448     A
MCG                SA1    UR                                                    MCG        2449     A
MCG                SX7    N.UR0                                                 MCG        2450     A
MCG                SX6    X1-1                                                  MCG        2451     A
MCG       +        IX3    X6-X7                                                 MCG        2452     A
MCG                PL     X3,*+1      UR = MAX( UR-1 , N.UR0 )                  MCG        2453     A
MCG                BX6    X7                                                    MCG        2454     A
MCG                SA6    A1                                                    MCG        2455     A
MCG                                                                             MCG        2456     A
MCG        JAM3    LX5    1+I.IIP                                               MCG        2457     A
MCG                RJ     AIL         ADJUST THE ISSUE LIST                     MCG        2458     A
MCG                RJ     DUC         DECREMENT USES COUNTS                     MCG        2459     A
MCG                S"DI"  "DI"                                                  MCG        2460     A
MCG                RJ     ASC         ADVANCE CLOCKS                            MCG        2461     A
MCG                SA5    "DI"-2                                                MCG        2462     A
MCG                RJ     SII         SAVE THE INSTRUCTION                      MCG        2463     A
MCG                                                                             MCG        2464     A
MCG        JAM4    S"DI"  "DI"                                                  MCG        2465     A
MCG                RJ     SNU         SET NEXT USE ADDRESS                      MCG        2466     A
MCG                                                                             MCG        2467     A
MCG        JAM5    ZR     X7,JAM1     IF UOR = 0    */ NO UNCLEARED OPERAND REGS MCG        2468     A
MCG                RJ     CNU         CHECK NEXT USES OF OPERANDS               MCG        2469     A
MCG                EQ     JAM1                                                  MCG        2470     A
MCG                                                                             MCG        2471     A
MCG        JAM6    SA3    ICL                                                   MCG        2472     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SX7    0                                                    MCG        2473    A
MCG                SA7    UOR        UOR = 0                                   MCG        2474    A
MCG                RJ     UJR        UNJAM THE REGISTERS                       MCG        2475    A
MCG                SA1    UOR                                                  MCG        2476    A
MCG                SA2    T0                                                   MCG        2477    A
MCG                SB6    "TB"-B1                                              MCG        2478    A
MCG                BX7    X1                                                   MCG        2479    A
MCG                S"DI"  B6+X2      DI = BASE-1 + T0                          MCG        2480    A
MCG                EQ     JAM5                                                 MCG        2481    A
MCG        JCG     SPACE  3                                                    MCG        2482    A
MCG                USE    /MCG/                                                MCG        2483    A
MCG        M.OT    BSS    1          MAX NUMBER OF OT.S IN CURRENT SEQUENCE    MCG        2484    A
MCG        SF      BSS    1          = FX.ST IF UNSAVED RESULTS NOT IN ST REGS MCG        2485    A
MCG        LRN     BSS    8          LAST R-NUMBER IN B-REGISTER I             MCG        2486    A
MCG        ISC     EQU    LRN        ISSUED STORE CHAIN                        MCG        2487    A
MCG                                                                            MCG        2488    A
MCG        CLASS   BSS    1          REGISTER CLASS                            MCG        2489    A
MCG        R0      BSS    1          *TXT* INDEX OF INSTRUCTION  REMOVED FROM REGS MCG    2490    A
MCG        T0      BSS    1          *TXT* INDEX OF INSTRUCTION TO BE ISSUED   MCG        2491    A
MCG        T0C     BSS    1          COPY OF T0                                MCG        2492    A
MCG        SIW     BSS    1          INDEX OF AVAIL WORD IN *SIT*              MCG        2493    A
MCG        DRI     BSS    4          DUMMY RLIST INSTRUCTION BUFFER            MCG        2494    A
MCG        UOR     BSS    1          LIST OF UNCLEARED OP REGISTERS ( *UJR* )  MCG        2495    A
MCG        UR      CON    9          COUNT OF UNLOADED REGISTERS               MCG        2496    A
MCG        PIS     BSS    1          POINT IN SEQUENCE                         MCG        2497    A
MCG        TM      BSS    1          MAX *LST* OF ISSUABLE INST TO CONSIDER    MCG        2498    A
MCG                USE    0                                                    MCG        2499    A
MCG        SNI     TITLE  SNI - SELECT NEXT INSTRUCTION TO ISSUE               MCG        2500    A
MCG       **       SNI - SELECT NEXT INSTRUCTION TO BE POSTED ( ISSUED ) FROM  MCG        2501    A
MCG       *        THE *ICL* ( ISSUE CANIDATE LIST ).                          MCG        2502    A
MCG       *        SCAN INSTRUCTIONS ON *ICL* FOR WHICH LST[INST] < LST[ICL] + 7 MCG      2503    A
MCG       *        OTHERWISE THIS ROUTINE IS IDENTICAL TO THE SCHEDULER        MCG        2504    A
MCG       *        VERSION OF *SNI*.                                           MCG        2505    A
MCG       *                                                                    MCG        2506    A
MCG       *        EXIT   (B3) = REGNO OF ISSUABLE INST, < 0 IF NONE FOUND     MCG        2507    A
MCG       *               (B4,B5) = T(EX), T(IS)                               MCG        2508    A
MCG       *               (A5) = LI ( ADDRESS OF LINK WORD OF ISSUABLE INSTRUCTION MCG   2509    A
MCG       *               (A0) = *TXT* INDEX OF PREVIOUS INSTRUCTION           MCG        2510    A
MCG       *               PBI WORDS SET.                                       MCG        2511    A
MCG                                                                            MCG        2512    A
MCG        SNI0    SA5    "DI"+1     LI = DI + 1                               MCG        2513    A
MCG                                                                            MCG        2514    A
MCG        SNI     ROUTINE                                                     MCG        2515    A
MCG                SA4    PBI-1                                                MCG        2516    A
MCG                SA3    ICL        CI = ICL                                  MCG        2517    A
MCG                S"PI"  B0         PI = 0                                    MCG        2518    A
MCG                BX6    X4                                                   MCG        2519    A
MCG                SA6    A4+B1      PBI = PB1(FALSE,0,INFINITY,INFINITY)      MCG        2520    A
MCG                SB6    X4         TIS[PBI] = INFINITY                       MCG        2521    A
MCG                SA5    "TB"+X3    L = BASE + CI                             MCG        2522    A
MCG                MX7    -I.LSTL                                              MCG        2523    A
MCG                LX5    -I.LSTP                                              MCG        2524    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                BX2    -X7*X5                                                  MCG        2525    A
MCG                SX6    X2+7        TM = LST[L] + 7  */ MAX LATE START TIME     MCG        2526    A
MCG                SA6    TM                                                      MCG        2527    A
MCG                                                                               MCG        2528    A
MCG      *         GET INST DESCR, CHECK FOR FIT IN CURRENT WORD                  MCG        2529    A
MCG                                                                               MCG        2530    A
MCG       SNI1     SB4    X3+"TB"     LI = CI + BASE                              MCG        2531    A
MCG                SA2    T                                                       MCG        2532    A
MCG                SA4    B4-B1       DI = LI - 1  ( DESCR)                       MCG        2533    A
MCG                SB5    X2          T(IS) = T                                   MCG        2534    A
MCG                LX4    58-D.SZP                                                MCG        2535    A
MCG                PL     X4,SNI4     IF SZ[DI] < 2 */ NO POSSIBLE SIZE FAULT     MCG        2536    A
MCG                SA3    PC                                                      MCG        2537    A
MCG                MX7    -2                                                      MCG        2538    A
MCG                BX6    X3+X7                                                   MCG        2539    A
MCG                NZ     X6,SNI4     IF PC " 3     */ NOT IN LAST PARCEL OF WORD MCG        2540    A
MCG                BX2    X4                                                      MCG        2541    A
MCG                LX2    1+D.SZP-D.SRP                                           MCG        2542    A
MCG                PL     X2,SNI3     IF ^SR[DI]    */ NOT SIZE REDUCABLE         MCG        2543    A
MCG                                                                               MCG        2544    A
MCG      *         INSTRUCTION IS SIZE REDUCEABLE, CHECK A-REGISTERS TO SEE IF    MCG        2545    A
MCG      *         ADDRESS IS IN ONE OF THEM.                                     MCG        2546    A
MCG                                                                               MCG        2547    A
MCG                SA2    ARA+7       I = ARA(8)                                  MCG        2548    A
MCG                SA3    "DI"-B1     R2 = DI - 1                                 MCG        2549    A
MCG                SB2    ARA-1                                                   MCG        2550    A
MCG                BX6    X3                                                      MCG        2551    A
MCG                SA6    B2+B1       ARA(1) = [R2]                               MCG        2552    A
MCG                                                                               MCG        2553    A
MCG       SNI2     BX7    X3-X2                                                   MCG        2554    A
MCG                SA2    A2-B1       I = I - 1                                   MCG        2555    A
MCG                NZ     X7,SNI2     IF [I+1] " [R2]                             MCG        2556    A
MCG                SB3    A2-B2                                                   MCG        2557    A
MCG                ZR     B3,SNI3     IF I = 0    */ MATCH WITH SEARCH TERMINATOR MCG        2558    A
MCG                                                                               MCG        2559    A
MCG                SA2    TRA+RN.A0+B3              */ COMPUTE TRUE ISSUE TIME    MCG        2560    A
MCG                SB2    X2                                                      MCG        2561    A
MCG                LE     B2,B5,SNI4                                              MCG        2562    A
MCG                SB5    X2                                                      MCG        2563    A
MCG                EQ     SNI4                                                    MCG        2564    A
MCG                                                                               MCG        2565    A
MCG       SNI3     SA1    IWT               */ SET PENALITY FOR SIZE FAULT        MCG        2566    A
MCG                SX2    X1+B1                                                   MCG        2567    A
MCG                SB5    X2+B5       T(IS) = T(IS) + IWT + 1                     MCG        2568    A
MCG                                                                               MCG        2569    A
MCG      *         SELECT AN AVAILABLE FUNCTIONAL UNIT                            MCG        2570    A
MCG                                                                               MCG        2571    A
MCG       SNI4     LX4    2+D.SZP-D.F1P                                           MCG        2572    A
MCG                MX6    -D.F1L                                                  MCG        2573    A
MCG                BX7    -X6*X4      F = F1[DI]                                  MCG        2574    A
MCG                SA3    TFU+X7      TFA = TFU(F)                                MCG        2575    A
MCG                SB3    X3                                                      MCG        2576    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                LE     B3,B5,SNI5  IF TFA @ T(IS)                                  MCG   2577  A
MCG        #MD     IFEQ   .CPU,74                                        ?6600        MCG   2578  A
MCG                LX4    D.F1P-D.F2P                                                 MCG   2579  A
MCG                SX1    X7                        */ CHECK SECOND FUNIT             MCG   2580  A
MCG                BX7    -X6*X4      F2 = F2[DI]                                     MCG   2581  A
MCG                SA2    TFU+X7      TF2 = TFU(F2)                                   MCG   2582  A
MCG                LX4    D.F2P-D.F1P                                                 MCG   2583  A
MCG                SB4    X2                                                          MCG   2584  A
MCG                LE     B4,B5,SNI5  IF TF2 @ T(IS)                                  MCG   2585  A
MCG                SB5    B4                                                          MCG   2586  A
MCG                LE     B4,B3,SNI5  IF TF2 @ TF1                                    MCG   2587  A
MCG                BX7    X1                                                          MCG   2588  A
MCG        #MD     ENDIF                                                 '6600        MCG   2589  A
MCG                SB5    X3          T(IS) = TF1   */ MIN(TFA,TF2)                   MCG   2590  A
MCG                                                                                   MCG   2591  A
MCG       *        IF T(IS) @ TIS[PBI] THEN TRY TO GET A RESULT REG, CALC T(EX)        MCG   2592  A
MCG                                                                                   MCG   2593  A
MCG        SNI5    GT     B5,B6,SNI6  IF T(IS) > TIS[PBI]                             MCG   2594  A
MCG                SA7    SNIA                                                        MCG   2595  A
MCG                LX4    D.F1P+59-D.NRP                                              MCG   2596  A
MCG                RJ     DRR         CALL DRR( DI , T(IS) , T(EX) , REGNO )          MCG   2597  A
MCG                GT     B3,SNI7     IF REGNO > 0  */ INSTRUCTION IS ISSUABLE        MCG   2598  A
MCG                ZR     B3,SNI0     IF REGNO = 0  */ A PSEUDO INSTRUCTION           MCG   2599  A
MCG                                                                                   MCG   2600  A
MCG        SNI6    SA5    "DI"+B1     LI = DI + 1                                     MCG   2601  A
MCG                EQ     SNI9                      */ NO RESULT REG AVAIL            MCG   2602  A
MCG                                                                                   MCG   2603  A
MCG       *        COMPARE THIS INSTRUCTION TIMING WITH THAT OF *PBI*                  MCG   2604  A
MCG                                                                                   MCG   2605  A
MCG        SNI7    SA5    "DI"+B1     LI = DI + 1                                     MCG   2606  A
MCG        #MD     IFEQ   .CPU,74                                        ?6600        MCG   2607  A
MCG                GT     B5,B6,SNI9  IF T(IS) > TIS[PBI]                             MCG   2608  A
MCG                SA3    PBI                                                         MCG   2609  A
MCG                LT     B5,B6,SNI8  IF T(IS) < TIS[PBI]                             MCG   2610  A
MCG                LX3    -PB.TEXP                                                    MCG   2611  A
MCG                SB2    X3                                                          MCG   2612  A
MCG                GE     B4,B2,SNI9  IF T(EX) \ TEX[PBI]                             MCG   2613  A
MCG        #MD     ELSE                                                  '66?76       MCG   2614  A
MCG                SA4    "DI"                                                        MCG   2615  A
MCG                GE     B4,B6,SNI9  IF T(EX) \ TEX[PBI]                             MCG   2616  A
MCG        #MD     IFGT   .CPU,74                                        ?7600        MCG   2617  A
MCG                LX4    -D.LDP                                                      MCG   2618  A
MCG        .D      BIT    D.^DP-D.LDP                                                 MCG   2619  A
MCG                SX6    .D+1                                                        MCG   2620  A
MCG                SB5    B4          T(IS) = T(EX)                                   MCG   2621  A
MCG                BX7    X6*X4                                                       MCG   2622  A
MCG                LX4    D.LDP-D.FTP                                                 MCG   2623  A
MCG                NZ     X7,SNI8     IF ^D[DI] ! LD[DI]                              MCG   2624  A
MCG                                                                                   MCG   2625  A
MCG       *        ADJUST T(EX) TO ACCOUNT FOR THE FUNIT TO REGISTER INPUT PATH        MCG   2626  A
MCG                                                                                   MCG   2627  A
MCG                SA2    T                                                           MCG   2628  A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SA1    TRS                                                  MCG        2629     A
MCG                MX6    -D.FTL                                               MCG        2630     A
MCG                BX7    -X6*X4                                               MCG        2631     A
MCG                IX3    X7-X2                                                MCG        2632     A
MCG                SB2    X3+B5                                                MCG        2633     A
MCG                SB4    B4-B1       T(EX) = T(EX) - 1                        MCG        2634     A
MCG                AX1    B2,X1       XSW = SHIFT(TRS, -(T(IS)+FT[DI]-T) )     MCG        2635     A
MCG                MX3    1                                                    CCG0157       4     A
MCG                BX1    -X3*X1                                               CCG0157       5     A
MCG                                                                            MCG        2636     A
MCG        SNI7A   LX1    -1          XSW = SHIFT( XSW , 59 )                  MCG        2637     A
MCG                SB4    B4+B1       T(EX) = T(EX) + 1                        MCG        2638     A
MCG                MI     X1,SNI7A    IF XSW < 0    */ RESULT STORED AT THIS TIME MCG     2639     A
MCG                SB5    B4          T(IS) = T(EX)                            MCG        2640     A
MCG                GE     B4,B6,SNI9  IF T(EX) \ TEX[PBI]                      MCG        2641     A
MCG        #MD     ENDIF                                              '7600    MCG        2642     A
MCG                                                                            MCG        2643     A
MCG       *        INSTRUCTION BETTER THAN PREVIOUS BEST, SAVE INFO IN *PBI*  MCG        2644     A
MCG                                                                            MCG        2645     A
MCG        SNI8    SX4    B5                                                   MCG        2646     A
MCG                SA1    SNIA                                                 MCG        2647     A
MCG                SX3    B4                                                   MCG        2648     A
MCG                LX4    PB.TISP                                              MCG        2649     A
MCG                PX6    B3,X4                                                MCG        2650     A
MCG                SX7    "PI"                                                 MCG        2651     A
MCG                LX3    PB.TEXP                                              MCG        2652     A
MCG                SB2    X1                                                   MCG        2653     A
MCG                SX2    A5-"TB"                                              MCG        2654     A
MCG                BX6    X3+X6                                                MCG        2655     A
MCG                SA1    T                                                    MCG        2656     A
MCG                LX7    PB.PIP                                               MCG        2657     A
MCG                LX2    PB.LIP                                               MCG        2658     A
MCG                PX2    B2,X2                                                MCG        2659     A
MCG                SA6    PBI         PBI(0) = PBII(REG,T(EX),T(IS))           MCG        2660     A
MCG                IX7    X7+X2                                                MCG        2661     A
MCG                SB2    X1+                                                  MCG        2662     A
MCG                SA7    A6+B1       PBI(1) = PBII2(FU,PI,LI)                 MCG        2663     A
MCG                                                                            MCG        2664     A
MCG                EQ     B4,B2,SNI   IF T(EX) = T  */ EXIT IF THIS INST IS EXEC MCG        2665     A
MCG                                                                            MCG        2666     A
MCG                SB6    B5          TIS[PBI] = T(IS)                         MCG        2667     A
MCG                                                                            MCG        2668     A
MCG       *        ADVANCE TO NEXT INSTRUCTION ON CHAIN                       MCG        2669     A
MCG                                                                            MCG        2670     A
MCG        SNI9    S"PI"  A5-"TB"     PI = LI - BASE                          MCG        2671     A
MCG                LX5    -I.LINKP                                             MCG        2672     A
MCG                SX3    X5          CI = LINK[LI]                            MCG        2673     A
MCG                ZR     X3,SNI10    IF CI = 0                                MCG        2674     A
MCG                                                                            MCG        2675     A
MCG       *        ONLY CONSIDER INSTRUCTIONS THAT ARE NEAR THE FIRST ON THE ISSUE MCG   2676     A
MCG       *        LIST, SO WE DO NOT GO TOO FAR ASTRAY FROM THE PREDETERMINED MCG       2677     A
MCG       *        ORDER.                                                     MCG        2678     A

          0        1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                                                                                  MCG        2679     A
MCG                 MX6    -I.LSTL                                                   MCG        2680     A
MCG                 SA5    "TB"+X3     LI = BASE + CI                                MCG        2681     A
MCG                 SA2    TM                                                        MCG        2682     A
MCG                 LX5    -I.LSTP                                                   MCG        2683     A
MCG                 BX7    -X6*X5                                                    MCG        2684     A
MCG                 IX4    X7-X2                                                     MCG        2685     A
MCG                 MI     X4,SNI1     IF LST[LI] < TM  */ INSTRUCTION IS NEAR T0    MCG        2686     A
MCG                                                                                  MCG        2687     A
MCG        SNI10    SA1    PBI                                                       MCG        2688     A
MCG                 SA1    PBI                                                       MCG        2689          I
          -FCC0448
MCG                 SA2    A1+B1                                                     MCG        2690     A
MCG                 UX3    B3,X1       REGNO = REG[PBI]                              MCG        2691     A
MCG                 LX2    -PB.LIP                                                   MCG        2692     A
MCG                 MI     B3,SNI      IF REGNO < 0  */ NO ISSUABLE INST FOUND       MCG        2693     A
MCG                                                                                  MCG        2694     A
MCG                 SA5    X2+"TB"     LI = LI[PBI] + BASE                           MCG        2695     A
MCG                 LX2    PB.LIP-PB.PIP                                             MCG        2696     A
MCG                 S"PI"  X2          PI = PI[PBI]                                  MCG        2697     A
MCG                 LX1    -PB.TISP                                                  MCG        2698     A
MCG                 SB5    X1          T(IS) = TIS[PBI]                              MCG        2699     A
MCG                 LX1    PB.TISP-PB.TEXP                                           MCG        2700     A
MCG                 SB4    X1          T(EX) = TEX[PBI]                              MCG        2701     A
MCG                 EQ     SNI                                                       MCG        2702     A
MCG        ISI      TITLE  ISI - ISSUE SELECTED INSTRUCTION                          MCG        2703     A
MCG       **        ISI - ISSUE SELECTED INSTRUCTION                                 MCG        2704     A
MCG       *                                                                          MCG        2705     A
MCG       *         ENTRY  (X2) = R , *TXT* INDEX OF INSTRUCTION                     MCG        2706     A
MCG       *                                                                          FCC0584       5     A
MCG       *         EXIT   (B3) < IF INSTRUCTION WAS NOT ISSUED                      MCG        2707     A
MCG       *                UOR SET                                                   FCC0584       6     A
MCG                                                                                  MCG        2708     A
MCG        ISI0     SA4    "DI"                                                      MCG        2709     A
MCG                 RJ     SNU         SET ADDR OF NEXT USES                         MCG        2710     A
MCG                 SA7    UOR                                                       FCC0584       7     A
MCG                                                                                  MCG        2711     A
MCG        ISI      ROUTINE                                                          MCG        2712     A
MCG                 SA3    ICL         NI = ICL                                      MCG        2713     A
MCG                 SB6    "TB"-B1                                                   MCG        2714          I
          -CCGA023
MCG                 SA5    "TB"+X2     LI = BASE + R                                 CCGA023      11     A
MCG                 S"PI"  B0          PI = 0                                        MCG        2715     A
MCG                 SA4    X2+B6       DI = BASE-1 + R                               MCG        2716          I
          -CCGA023
MCG                 S"DI"  A5-B1       DI = LI - 1                                   CCGA023      12     A
MCG                 LX5    59-I.ICP                                                  CCGA023      13     A
MCG                 IX6    X2-X3                                                     MCG        2717     A
MCG                 PL     X5,*+1S17   IF ^IC[LI] BLOWUP                             CCGA023      14     A
MCG                 ZR     X6,ISI2     IF R = ICL                                    MCG        2718     A
MCG                                                                                  MCG        2719     A
MCG        ISI1     SA5    "TB"+X3     L = BASE + NI                                 MCG        2720     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                 S"PI"  X3          PI = NI                                  MCG        2721     A
MCG                 SX3    X5          NI = LINK[L]                             MCG        2722     A
MCG                 IX6    X3-X2                                                MCG        2723     A
MCG                 NZ     X6,ISI1     IF R " NI                                MCG        2724     A
MCG                                                                             MCG        2725     A
MCG        ISI2     SA1    T                                                    MCG        2726     A
MCG                 LX4    -D.F1P                                               MCG        2727     A
MCG                 MX5    -D.F1L                                               MCG        2728     A
MCG                 SB5    X1          T(IS) = T                                MCG        2729     A
MCG                 BX6    -X5*X4                                               MCG        2730     A
MCG                 SB2    X6          FU[PBI+1] = F1[DI]                       MCG        2731     A
MCG                 PX6    B2,X2       LI[PBI+1] = R                            MCG        2732     A
MCG                 LX4    D.F1P+59-D.NRP                                       MCG        2733     A
MCG                 SA6    PBI+1                                                MCG        2734     A
MCG                 RJ     DRR         REGNO = DRR( DI )                        MCG        2735     A
MCG                 ZR     B3,ISI0     IF REGNO = 0  */ A PSEUDO INSTRUCTION    MCG        2736     A
MCG                 MI     B3,ISI      IF REGNO < 0  */ NO RESULT REG FOUND     MCG        2737     A
MCG                 SX3    B5                                                   MCG        2738     A
MCG                 LX3    PB.TISP     TIS[PBI] = T(IS)                         MCG        2739     A
MCG                 SX2    B4                                                   MCG        2740     A
MCG                 SA5    "DI"+B1     L = DI + 1                               MCG        2741     A
MCG                 LX2    PB.TEXP                                              MCG        2742     A
MCG                 PX3    B3,X3       REG[PBI] = REGNO                         MCG        2743     A
MCG                 BX6    X2+X3       TEX[PBI] = T(EX)                         MCG        2744     A
MCG                 SA6    PBI                                                  MCG        2745     A
MCG                 LX5    59-I.IIP                                             MCG        2746     A
MCG                 PL     X5,ISI3     IF ^II[L]     */ NOT PREVIOUSLY ISSUED   MCG        2747     A
MCG                 SA1    UR                                                   MCG        2748     A
MCG                 SX7    N.UR0                                                MCG        2749     A
MCG                 SX6    X1-1                                                 MCG        2750     A
MCG       +         IX3    X6-X7                                                MCG        2751     A
MCG                 PL     X3,*+1      UR = MAX( UR-1 , N.UR0 )                 MCG        2752     A
MCG                 BX6    X7                                                   MCG        2753     A
MCG                 SA6    A1                                                   MCG        2754     A
MCG                                                                             MCG        2755     A
MCG        ISI3     LX5    1+I.IIP                                              MCG        2756     A
MCG                 RJ     AIL         ADVANCE THE ISSUE LIST                   MCG        2757     A
MCG                 RJ     DUC         DECREMENT USES COUNTS                    MCG        2758     A
MCG                 SA4    "DI"                                                 MCG        2759     A
MCG                 RJ     ASC         ADVANCE CLOCKS                           MCG        2760     A
MCG                 SA5    "DI"-2                                               MCG        2761     A
MCG                 RJ     SII         SAVE INSTRUCTION                         MCG        2762     A
MCG                 EQ     ISI0                                                 MCG        2763     A
MCG        CNU      TITLE  CNU - CHECK NEXT USE OF RESULTS IN REGISTERS         MCG        2764     A
MCG       **        CNU - CHECK NEXT USE OF RESULTS IN REGISTERS OF OPERANDS    MCG        2765     A
MCG       *                OF THE ISSUED INSTRUCTION AND RELEASE THE REGISTER   MCG        2766     A
MCG       *                IF THEIR NEXT USE IS "FAR AWAY".                     MCG        2767     A
MCG       *                                                                     MCG        2768     A
MCG       *         ENTRY  (X7) = UOR = LIST OF UNCLEARED OPERAND REGISTERS     MCG        2769     A
MCG       *                    UOR = 24/0,18/REG2,18/REG1,  REG1 " REG2 & REG1 " 0   MCG   2770     A
MCG       *                (A4) = DI OF INSTRUCTION                             MCG        2771     A
MCG       *                (UR) = COUNT OF UNLOADED REGISTERS                   MCG        2772     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG       *                                                                  MCG          2773     A
MCG       *         USES    B - 2, 3, 4, 5, 6  X - 1, 2, 3, 4, 5, 6, 7       MCG          2774     A
MCG                                                                          MCG          2775     A
MCG        MIN.D    EQU     14*4        MIN DISTANCE FOR UNLOAD CHECK ( 14 INST )  MCG    2776     A
MCG                                                                          MCG          2777     A
MCG        CNU0     SX6     0                                               FCC1580X        6     A
MCG                 SA6     UOR         UOR = 0                             FCC1580X        7     A
MCG        CNU      ROUTINE                                                 MCG          2778     A
MCG                 SA4     "DI"                                            FCC1580X        8     A
MCG                 LX4     59-D.PIP                                        FCC1580X        9     A
MCG                 MI      X4,CNU0     IF PI[DI]     */ *DEF* OR *RS*      FCC1580X       10     A
MCG                 SA5     "DI"+B1     L = DI + 1                          MCG          2779     A
MCG                 MX4     -I.LSTL                                         MCG          2780     A
MCG                 LX5     -I.LSTP                                         MCG          2781     A
MCG                 BX3     -X4*X5                                          MCG          2782     A
MCG                 AX6     B1,X3       PIS = (LST[L]/2)*4  */ POINT IN SEQ MCG          2783     A
MCG                 LX6     2                                               MCG          2784     A
MCG                 SA6     PIS                                             MCG          2785     A
MCG                                                                         MCG          2786     A
MCG       *         ORDER REG1,REG2 IN DECREASING DISTANCE OF NEXT USE      MCG          2787     A
MCG                                                                         MCG          2788     A
MCG                 BX3     X7                                              MCG          2789     A
MCG                 AX3     18                                              MCG          2790     A
MCG                 ZR      X3,CNU1     IF REG2 = 0                         MCG          2791     A
MCG                 SA1     RVT+X7                                          MCG          2792     A
MCG                 SA2     RVT+X3                                          MCG          2793     A
MCG                 LX1     -RV.NUP                                         MCG          2794     A
MCG                 LX2     -RV.NUP                                         MCG          2795     A
MCG                 SB2     X1                                              MCG          2796     A
MCG                 SB3     X2                                              MCG          2797     A
MCG                 GE      B2,B3,CNU1  IF NU[RVT(REG1)] \ NU[RVT(REG2)]    MCG          2798     A
MCG                 SX7     X7                                              MCG          2799     A
MCG                 LX7     18                                              MCG          2800     A
MCG                 BX7     X3+X7       SWAP( REG1 , REG2 )                 MCG          2801     A
MCG                                                                         MCG          2802     A
MCG        CNU1     MX1     -D.REGWL                                        MCG          2803     A
MCG                 SA2     UR                                              MCG          2804     A
MCG                 LX1     D.REGWP                                         MCG          2805     A
MCG                 SA5     PIS                                             MCG          2806     A
MCG                 LX2     D.REGWP                                         MCG          2807     A
MCG                 SB6     "TB"-B1                                         MCG          2808     A
MCG                 SB5     X5+MIN.D    TMIN = PIS + MIN.D                  MCG          2809     A
MCG                                                                         MCG          2810     A
MCG        CNU2     SA5     RVT+X7                                          MCG          2811     A
MCG                 SX6     X7-RN.X0                                        MCG          2812     A
MCG                 SB3     X7          REGNO = REG[UOR]                    MCG          2813     A
MCG                 AX7     18          UOR = SHIFT( UOR , -18 )            MCG          2814     A
MCG                 BX3     X5+X6                                           MCG          2815     A
MCG                 LX5     -RV.NUP                                         MCG          2816     A
MCG                 MI      X3,CNU4     IF LOCK[RVT(REGNO)] ! REGNO < RN.X0 MCG          2817     A
MCG                 SB4     X5          NS = NU[RVT(REGNO)]  */ POINT OF NEXT USE  MCG   2818     A
MCG                 SA4     B6+X5       DI = BASE-1 + NS                    MCG          2819     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                LE     B4,B5,CNU4  IF NS @ TMIN  */ USE IS NEARBY                MCG        2820     A
MCG                                                                                 MCG        2821     A
MCG      *         SEARCH BACKWARDS FROM THE POINT OF NEXT USE FOR A "PEAK" IN      MCG        2822     A
MCG      *         THE REG WIDTH"S, AND IF FOUND TRY TO DUMP THE REGISTER.          MCG        2823     A
MCG                                                                                 MCG        2824     A
MCG       CNU3     BX5    -X1*X4                                                    MCG        2825     A
MCG                IX6    X2-X5       N = UR - REGW[DI]                             MCG        2826     A
MCG                MI     X6,CNU5     IF N < 0      */ USE IS PAST A REG PEAK       MCG        2827     A
MCG                AX6    D.REGWP-2                                                 MCG        2828     A
MCG                SB2    X6+4        N = 4*(N+1)   */ MIN DIST TO MAKE REGW > UR   MCG        2829     A
MCG                SB4    B4-B2       NS = NS - N                                   MCG        2830     A
MCG                SA4    A4-B2       DI = DI - N                                   MCG        2831     A
MCG                GE     B4,B5,CNU3  IF NS \ TMIN                                  MCG        2832     A
MCG                                                                                 MCG        2833     A
MCG       CNU4     NZ     X7,CNU2     IF UOR " 0                                    MCG        2834     A
MCG                EQ     CNU                                                       MCG        2835     A
MCG                                                                                 MCG        2836     A
MCG      *         POINT OF NEXT USE OVER A REG WIDTH PEAK, DUMP THE REG            MCG        2837     A
MCG                                                                                 MCG        2838     A
MCG       CNU5     SA3    T                                                         MCG        2839     A
MCG                SA7    UOR                                                       MCG        2840     A
MCG                BX6    X3                                                        MCG        2841     A
MCG                SA6    TRA+B3      TRA(REGNO) = T                                MCG        2842     A
MCG                RJ     CRC         COMPUTE RESTORE COSTS FOR THIS REG            MCG        2843     A
MCG                RJ     SSR         SAVE THE RESULT                               MCG        2844     A
MCG                SA1    UOR                                                       MCG        2845     A
MCG                BX7    X1          UOR = CNUA                                    MCG        2846     A
MCG                ZR     X1,CNU      IF UOR = 0                                    MCG        2847     A
MCG                EQ     CNU1                                                      MCG        2848     A
MCG       SSR      TITLE  SSR - SAVE SPECIFIED REGISTER                             MCG        2849     A
MCG      **        SSR - SAVE SPECIFIED REGISTER                                    MCG        2850     A
MCG      *                                                                          MCG        2851     A
MCG      *         ENTRY  (B3) = REGNO , REGISTER TO BE FREED                       MCG        2852     A
MCG      *                (B4) = COST ( OF SAVING RESULT IN REGNO )                 MCG        2853     A
MCG                                                                                 MCG        2854     A
MCG       SSR0     SA1    UR                                                        MCG        2855     A
MCG                SX6    X1+B1       UR = UR + 1   */ INCREMENT COUNT OF UNL REGS  MCG        2856     A
MCG                SA6    A1                                                        MCG        2857     A
MCG                RJ     AIS         ALLOCATE INSTRUCTION SPACE                    MCG        2858     A
MCG                                                                                 MCG        2859     A
MCG       SSR      ROUTINE                                                          MCG        2860     A
MCG                SX6    B4-3                                                      MCG        2861     A
MCG                MI     X6,SSR3     IF COST < 3                                   MCG        2862     A
MCG                                                                                 MCG        2863     A
MCG      *         COST = 3, COMPUTATION NOT IN A STORE REG                         MCG        2864     A
MCG                                                                                 MCG        2865     A
MCG                SA1    AXR                                                       MCG        2866     A
MCG                SX7    FX.ST                                                     MCG        2867     A
MCG                BX6    X7*X1                                                     MCG        2868     A
MCG                NZ     X6,SSR1     IF AXR & FX.ST " 0  */ IF A STORE REG IS AVAIL MCG        2869     A
MCG                SX3    B1                                                        MCG        2870     A
MCG                SA2    RVT+B3                                                    MCG        2871     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                LX3    RV.SROP                                                MCG        2872    A
MCG                BX6    X3+X2       SRO[RVT(REGNO)] = 1                        MCG        2873    A
MCG                SA7    SF          SF = FX.ST    */ INDICATE UNSAVED RESULTS  MCG        2874    A
MCG                SA6    A2                                                     MCG        2875    A
MCG                EQ     SSR                                                    MCG        2876    A
MCG                                                                              MCG        2877    A
MCG        SSR1    SB4    B3+         REGF = REGNO                               MCG        2878    A
MCG                SB3    RN.X0+6                                                MCG        2879    A
MCG                SB2    OC.XMT                                                 MCG        2880    A
MCG                SA3    RVT+B3                                                 MCG        2881    A
MCG                ZR     X3,SSR2     REGR = IF( RVT(X6) = 0 ) THEN X6; ELSE X7  MCG        2882    A
MCG                SB3    B3+1                                                   MCG        2883    A
MCG        SSR2    RJ     OXI         MOVE RESULT TO A STORE REG                 MCG        2884    A
MCG                SB4    2           COST = 2                                   MCG        2885    A
MCG                                                                              MCG        2886    A
MCG        SSR3    SA1    RVT+B3                                                 MCG        2887    A
MCG                BX6    "RN"X1      R0 = RI[RVT(REGNO)]                        MCG        2888    A
MCG                BX1    X6                                                     MCG        2889    A
MCG                SA6    R0                                                     MCG        2890    A
MCG                PRINT  SSR,(* R0,UR = *,O6,I5),(R0,UR)                        MCG        2891    A
MCG                SA2    AXR                                                    MCG        2892    A
MCG                SA3    RBV+B3                                                 MCG        2893    A
MCG                MX6    0                                                      MCG        2894    A
MCG                BX7    X2+X3       AXR = AXR ! RBV(REGNO)  */ MAKE THE REG AVAIL MCG        2895    A
MCG                SA6    A1          RVT(REGNO) = 0                             MCG        2896    A
MCG                SA7    A2                                                     MCG        2897    A
MCG                                                                              MCG        2898    A
MCG                GT     B4,B1,SSR4  IF COST > 1                                MCG        2899    A
MCG                RJ     RII         REISSUE THE INSTRUCTION                    MCG        2900    A
MCG                EQ     SSR0                                                   MCG        2901    A
MCG                                                                              MCG        2902    A
MCG        SSR4    SX6    0                                                      MCG        2903    A
MCG                SA6    T0          T0 = 0                                     MCG        2904    A
MCG                RJ     SRM         SAVE RESULT IN MEMORY                      MCG        2905    A
MCG                EQ     SSR0                                                   MCG        2906    A
MCG        SUR     SPACE  3,14        SUR                                        MCG        2907    A
MCG       **       SUR - STORE UNSAVED RESULTS                                   MCG        2908    A
MCG       *                                                                      MCG        2909    A
MCG       *        ENTRY  AXR & FX. ST " 0 ( STORE REGISTER FREE )               MCG        2910    A
MCG       *                SF " 0 , UNSAVED RESULTS IN THE REGISTERS THAT       MCG        2911    A
MCG       *                     HAVE NO IMMEDIATE USES.                         MCG        2912    A
MCG                                                                              MCG        2913    A
MCG        SUR     ROUTINE                                                       MCG        2914    A
MCG                SB3    RN.X0+6                                                MCG        2915    A
MCG                SB4    RN.X0       I = RN.X0                                  MCG        2916    A
MCG                SA1    RVT+B3                                                 MCG        2917    A
MCG                SX6    0           T0 = 0                                     MCG        2918    A
MCG                SA6    T0                                                     MCG        2919    A
MCG                ZR     X1,SUR1     REGR = IF( RVT(X6) = 0 ) THEN X6; ELSE X7  MCG        2920    A
MCG                SB3    B3+1                                                   MCG        2921    A
MCG                                                                              MCG        2922    A
MCG        SUR1    SA1    RVT+B4                                                 MCG        2923    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                BX7    "RN"X1                                                  MCG        2924     A
MCG                LX1    59-RV.SROP                                              MCG        2925     A
MCG                PL     X1,SUR2      IF ^SRO[RVT(I)]                            MCG        2926     A
MCG                                                                               MCG        2927     A
MCG                SB2    OC.XMT                                                  MCG        2928     A
MCG                SA7    R0           R0 = RI[RVT(I)]                            MCG        2929     A
MCG                SX2    B4                                                      MCG        2930     A
MCG                PX6    B3,X2                                                   MCG        2931     A
MCG                SA6    SURA                                                    MCG        2932     A
MCG                RJ     OXI          OUTPUT AND XMIT                            MCG        2933     A
MCG                SA1    UR                                                      MCG        2934     A
MCG                SX6    X1+B1        UR = UR + 1                                MCG        2935     A
MCG                SA6    A1                                                      MCG        2936     A
MCG                RJ     SRM          AND SAVE RESULT IN MEMORY                  MCG        2937     A
MCG                SA2    SURA                                                    MCG        2938     A
MCG                SB4    X2                                                      MCG        2939     A
MCG                UX6    B3,X2                                                   MCG        2940     A
MCG                                                                               MCG        2941     A
MCG        SUR2    SB4    B4+B1        I = I + 1                                  MCG        2942     A
MCG                SX6    B4-RN.X0-6                                              MCG        2943     A
MCG                MI     X6,SUR1      IF I < RN.X6                               MCG        2944     A
MCG                                                                               MCG        2945     A
MCG                SA1    AXR                                                     MCG        2946     A
MCG                SA2    RBV+B3                                                  MCG        2947     A
MCG                SA6    RVT+B3       RVT(REGR) = 0 */ MAKE REG AVAIL            MCG        2948     A
MCG                SA6    SF           SF = 0                                     MCG        2949     A
MCG                BX7    X1+X2        AXR = AXR ! RBV(REGR)                      MCG        2950     A
MCG                SA7    A1                                                      MCG        2951     A
MCG                EQ     SUR                                                     MCG        2952     A
MCG                                                                               MCG        2953     A
MCG         SURA   BSS    1                                                       MCG        2954     A
MCG         UJR    TITLE  UJR - UNJAM REGISTERS                                   MCG        2955     A
MCG        **      UJR - UNJAM REGISTERS, FORCE ISSUE OF CURRENT INSTRUCTION      MCG        2956     A
MCG        *              BY FREEING UP A REGISTER, CREATING AN *XMT*, ETC.      MCG        2957     A
MCG        *                                                                      MCG        2958     A
MCG        *       ENTRY  (X3) = T0 , *TXT* INDEX OF INSTRUCTION TO BE ISSUED     MCG        2959     A
MCG        *                                                                      MCG        2960     A
MCG        *       EXIT   (UOR) = EXIT VALUE FROM *SNU* IF *ISI* WAS CALLED       MCG        2961     A
MCG                                                                               MCG        2962          I
           -FCC0584
MCG                                                                               MCG        2963     A
MCG         UJR    ROUTINE                                                        MCG        2964     A
MCG                SB6    "TB"-1                                                  MCG        2965     A
MCG                S"DI"  X3+B6        DI = BASE-1 + T0                           MCG        2966     A
MCG                BX6    X3                                                      MCG        2967     A
MCG                SA6    T0C          T0C = T0                                   MCG        2968     A
MCG                SA6    T0                                                      MCG        2969     A
MCG                RJ     DRT          CLASS = DRT( DI ) */ DETERMINE REG CLASS   MCG        2970     A
MCG                PRINT  UJR,(* TO = *,O6),T0                                    MCG        2971     A
MCG                PRNT   UJR,RUI                                                 MCG        2972     A
MCG                SA6    CLASS                                                   MCG        2973     A
MCG                RJ     AIS          ALLOCATE INSTRUCTION SPACE                 MCG        2974          I

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

          -FCC1451
MCG                 RJ     CSI         CHECK STORE REGISTERS                          MCG         2975          I
          -FCC1451
MCG                 SX7    0                                                          FCC1451        8     A
MCG                 SA7    ISI         ISI = 0       */ INDICATE *ISI* NOT ENTERED    FCC1451        9     A
MCG                 RJ     CSI         CHECK STORE REGISTERS                          FCC1451       10     A
MCG                 SA1    ISI                                                        FCC1451       11     A
MCG                 ZR     X1,UJR0     IF ISI = 0    */ NO INSTRUCTIONS ISSUED        FCC1451       12          I
          -CCGA019
MCG                 ZR     X1,UJR0A    IF ISI = 0  */NO INSTRUCTIONS ISSUED           CCGA019       10     A
MCG                                                                                   FCC1451       13     A
MCG                 SA2    T0                                                         FCC1451       14          I
          -CCGA019
MCG        UJR0     SA2    T0                                                         CCGA019       11     A
MCG                 RJ     ISI         TRY AND ISSUE T0                               FCC1451       15     A
MCG                 PRINT  JAM,(* CSI ISSUED A ST, REGNO =*Z2),B3                     FCC1451       16     A
MCG                 PL     B3,UJR      IF REGNO \ 0  */ ISI ISSUED THE INSTRUCTION    FCC1451       17     A
MCG                                                                                   FCC1451       18     A
MCG        UJR0     RJ     AIS         ALLOCATE INSTRUCTION SPACE                     FCC1451       19          I
          -CCGA019
MCG        UJR0A    RJ     AIS         ALLOCATE INSTRUCTION SPACE                     CCGA019       12     A
MCG                 SA1    AXR                                                        MCG         2976     A
MCG                 SX4    FX.SCR                                                     MCG         2977     A
MCG                 BX6    X4*X1       AVAIL = AXR & FX.SCR                           MCG         2978     A
MCG                 ZR     X6,UJR2     IF AVAIL = 0                                   MCG         2979     A
MCG                 SA1    CLASS                                                      MCG         2980     A
MCG                 BX7    X1*X6                                                      MCG         2981     A
MCG                 NZ     X7,UJR      IF AVAIL & CLASS " 0                           MCG         2982     A
MCG                                                                                   MCG         2983     A
MCG        UJR1     SA1    CLASS                                                      MCG         2984     A
MCG                 RJ     CRA         CHANGE REGISTER ASSIGNMENTS                    MCG         2985          I
          -FCC0584
MCG                 RJ     CRA         CHANGE REGISTER ASSIGNMENTS AND ISSUE *T0*     FCC0584        8     A
MCG                 EQ     UJR                                                        MCG         2986     A
MCG                                                                                   MCG         2987     A
MCG        UJR2     SA1    CLASS                                                      MCG         2988     A
MCG                 RJ     FRR         REGF = FRR(RVT,MCOST)  */ FIND A RESULT REG    MCG         2989     A
MCG                 SX7    B4-3                                                       MCG         2990     A
MCG                 MX5    0                                                          CCGA019       13     A
MCG                 MI     X7,UJR3     IF MCOST < 3                                   MCG         2991     A
MCG                 SA1    T0                                                         CCG0110        7     A
MCG                 SB6    "TB"-B1                                                    CCG0110        8     A
MCG                 S"DI"  B6+X1                                                      CCG0110        9     A
MCG                 MX5    D.TYL                                                      CCG0110       10     A
MCG                 BX6    X5*X4                                                      CCG0110       11     A
MCG                 NZ     X6,UJR2A    IF TY[DI] NE TYPE_I                            CCG0110       12     A
MCG                 LX4    59-D.PRSP                                                  CCG0110       13     A
MCG                 PL     X4,UJR2A    IF NOT PRS[DI]                                 CCG0110       14     A
MCG                 RJ     IIX         ISSUE INSTRUCTION AND XMT"S                    CCG0110       15     A
MCG                 ZR     X3,UJR      IF ISSUED                                      CCG0110       16     A
MCG                                                                                   MCG         2992     A
MCG       *         DUMP THE STORE REG WHOSE FIRST USE IS FURTHEST AWAY               MCG         2993     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                                                                                  MCG          2994     A
MCG                 SA5    O.TREE                                                    MCG          2995          I
          -CCG19
MCG                 SB3    RN.X0+6     REGF = X6                                     MCG          2996          I
          -CCG19
MCG                 SA1    RVT+B3                                                    MCG          2997          I
          -CCG19
MCG                 MX7    -I.INDXL                                                  MCG          2998          I
          -CCG19
MCG                 SA2    A1+B1                                                     MCG          2999          I
          -CCG19
MCG                 BX6    X1*X2                                                     FCC0881        32          I
          -CCG19
MCG                 MI     X6,UJR.E1   IF LOCK[RVT(X6)] & LOCK[RVT(X7)]              FCC0881        33          I
          -CCG19
MCG                 SB2    X5                                                        MCG          3000          I
          -CCG19
MCG                 SA3    "TB"+X1     RX6 = BASE + RI[RVT(X6)]                      MCG          3001          I
          -CCG19
MCG                 SA4    "TB"+X2     RX7 = BASE + RI[RVT(X7)]                      MCG          3002          I
          -CCG19
MCG                 LX3    -I.INDXP                                                  MCG          3003          I
          -CCG19
MCG                 LX4    -I.INDXP                                                  MCG          3004          I
          -CCG19
MCG                 BX6    -X7*X3                                                    MCG          3005          I
          -CCG19
MCG                 BX7    -X7*X4                                                    MCG          3006          I
          -CCG19
MCG                 SA3    B2+X6       UI6 = O.TREE + INDX[RX6]                      MCG          3007          I
          -CCG19
MCG                 SA4    B2+X7       UI7 = O.TREE + INDX[RX7]                      MCG          3008          I
          -CCG19
MCG                 LX3    -U.SUCCP                                                  MCG          3009          I
          -CCG19
MCG                 LX4    -U.SUCCP                                                  MCG          3010          I
          -CCG19
MCG                 AX1    59                                                        FCC0881        34          I
          -CCG19
MCG                 AX2    59                                                        FCC0881        35          I
          -CCG19
MCG                 BX3    -X1*X3                                                    FCC0881        36          I
          -CCG19
MCG                 BX4    -X2*X4                                                    FCC0881        37          I
          -CCG19
MCG                 SB5    X3                                                        MCG          3011          I
          -CCG19
MCG                 SB6    X4                                                        MCG          3012          I
          -CCG19
MCG                 SB4    B1+B1       MCOST = 2                                     MCG          3013          I
          -CCG19
MCG       +         GT     B5,B6,*+1   IF( SUCC[UI6] > SUCC[UI7] )                   MCG          3014          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

          -CCG19
MCG                SB3    B3+1              THEN X6; ELSE X7                    MCG        3015         I
          -CCG19
MCG                RJ     FSR         FIND A STORE REGISTER                    CCG19         1         I
          -CCG0110
MCG        UJR2A   RJ     FSR         FIND A STORE REG                         CCG0110      17    A
MCG                ZR     B3,UJR.E1   IF REGNO = 0  */ NOTHING FOUND           CCG19         2         I
          -CCGA019
MCG                                                                            MCG        3016    A
MCG        UJR3    SA1    RVT+B3                                               MCG        3017    A
MCG                SA2    RVT+RN.X0+6                                          CCGA023      15         I
          -CCGA031
MCG                SA3    A2+B1                                                CCGA023      16         I
          -CCGA031
MCG                BX6    "RN"X1      R0 = RI[RVT(REGF)]                       MCG        3018    A
MCG                BX5    X2*X3                                                CCGA023      17         I
          -CCGA031
MCG                SA6    R0                                                   MCG        3019    A
MCG                PRINT  UJR,(* R0 = *,O6),R0                                 MCG        3020    A
MCG                SA1    AXR                                                  MCG        3021    A
MCG                SA2    RBV+B3                                               MCG        3022    A
MCG                SA3    CLASS                                                MCG        3023    A
MCG                BX6    X1+X2       AXR = AXR ! RBV(REGF) */ MAKE REG AVAIL  MCG        3024    A
MCG                SA6    A1                                                   MCG        3025    A
MCG                MX7    0                                                    MCG        3026    A
MCG                SA7    RVT+B3                                               CCGA019      14    A
MCG                MI     X5,UJR10    IF BOTH X6 AND X7 LOCKED                 CCGA019      15    A
MCG                BX5    X2*X3                                                MCG        3027    A
MCG                SA7    RVT+B3      RVT(REGF) = 0                            MCG        3028         I
          -CCGA019
MCG                NZ     X5,UJR5     IF CLASS & RBV(REGF) " 0                 MCG        3029    A
MCG                                                                            MCG        3030    A
MCG       *        REGF NOT IN *CLASS*, FREE IT UP AND OUTPUT AN *XMT*         MCG        3031    A
MCG                                                                            MCG        3032    A
MCG                LE     B4,B1,UJR4  IF MCOST @ 1                             MCG        3033    A
MCG                RJ     TGL         SHOULD R0 BE TOGGLED                     CCGA023      18    A
MCG                NZ     B4,UJR0     IF B4 " 0 R0 WAS TOGGLED                 CCGA023      19    A
MCG                SX6    0                                                    MCG        3034    A
MCG                SA6    T0          T0 = 0                                   MCG        3035    A
MCG                RJ     SRM         SAVE RESULT IN MEMORY                    MCG        3036    A
MCG                SA5    T0C                                                  MCG        3037    A
MCG                SX6    X5+         T0 = T0C      */ RESTORE T0              MCG        3038    A
MCG                SA6    T0                                                   MCG        3039    A
MCG                EQ     UJR1                                                 MCG        3040    A
MCG                                                                            MCG        3041    A
MCG        UJR4    SA1    R0                                                   MCG        3042    A
MCG                RJ     RII         RESET FOR REISSUE                        MCG        3043    A
MCG                EQ     UJR1                                                 MCG        3044    A
MCG                                                                            MCG        3045    A
MCG       *        REGF IN *CLASS*, FREE IT UP                                 MCG        3046    A
MCG                                                                            MCG        3047    A
MCG        UJR5    LE     B4,B1,UJR6  IF MCOST @ 1                             MCG        3048    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                                                                                  MCG        3049     A
MCG                RJ     TGL         SHOULD R0 BE TOGGLED                           CCGA023      20     A
MCG                NZ     B4,UJR0     IF B4 " 0 R0 WAS TOGGLED                       CCGA023      21     A
MCG                RJ     SRM         SAVE RESULT IN MEMORY                          MCG        3050     A
MCG                EQ     UJR                                                        MCG        3051     A
MCG                                                                                 MCG        3052     A
MCG       UJR6     SA2    T0                                                         MCG        3053     A
MCG                RJ     ISI         ISSUE T0                                       MCG        3054     A
MCG                SA1    R0                                                         MCG        3055     A
MCG                SA7    UOR                                                        MCG        3056          I
          -FCC0584
MCG                RJ     RII         RESET FOR REISSUE                              MCG        3057     A
MCG                EQ     UJR                                                        MCG        3058     A
MCG                                                                                 CCGA019      16     A
MCG       *        BOTH X6 AND X7 LOCKED UP - FREE ONE OF THEM.                       CCGA019      17     A
MCG                                                                                 CCGA019      18     A
MCG       UJR10    SA1    R0                                                         CCGA019      19     A
MCG                SA3    LXR                                                        CCGA019      20     A
MCG                SB2    "TB"-1                                                     CCGA019      21     A
MCG                BX7    -X2*X3                                                     CCGA019      22     A
MCG                S"DI"  B2+X1       DI OF R0                                       CCGA019      23          I
          -CCG0008
MCG                SA7    A3          CLEAR LOCKOUT BIT FOR R0                       CCGA019      24          I
          -CCG0008
MCG                LX4    59-D.PRSP                                                  CCGA019      25          I
          -CCG0008
MCG                PL     X4,UJR.E1   IF ^PRS ABORT                                  CCGA019      26          I
          -CCG0008
MCG                SA7    A3          CLEAR LOCKOUT BIT FOR R0                       CCG0008       6     A
MCG                SX6    B3                                                         CCG0008       7     A
MCG                SA6    UJRA        UJRA = REGNO                                   CCG0008       8     A
MCG                                                                                 CCGA019      27     A
MCG                RJ     OTS         OUTPUT TEMP STORE                              CCGA019      28     A
MCG                                                                                 CCGA019      29     A
MCG                SA1    R0                                                         CCGA019      30     A
MCG                RJ     GLI         GENERATE LOAD OF TEMP                          CCGA019      31     A
MCG                                                                                 CCGA019      32     A
MCG       *        UPDATE DESCRIPTOR AND LINK WORDS OF R0.                            CCGA019      33          I
          -CCG0008
MCG                                                                                 CCGA019      34          I
          -CCG0008
MCG                SA1    R0                                                         CCGA019      35     A
MCG                SA3    "TB"+X1                                                    CCGA019      36     A
MCG                SA4    A3-B1       DI = LI - 1                                    CCG0008       9     A
MCG                BX7    X4                                                         CCG0008      10     A
MCG                LX7    59-D.PRSP                                                  CCG0008      11     A
MCG                PL     X7,UJR40    IF ^PRS[DI]                                    CCG0008      12     A
MCG                                                                                 CCG0008      13     A
MCG       *        UPDATE DESCRIPTOR AND LINK WORDS OF R0.                            CCG0008      14     A
MCG                                                                                 CCG0008      15     A
MCG                SX2    B1                                                         CCGA019      37     A
MCG                LX3    -I.USESP                                                   CCGA019      38     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                IX7    X2+X3       USES = USES + 1                          CCGA019      39     A
MCG                SA4    A3-B1       DI = LI - 1                              CCGA019      40          I
          -CCG0008
MCG                LX7    I.USESP                                              CCGA019      41     A
MCG                SA7    A3                                                   CCGA019      42     A
MCG                LX2    D.PRSP                                               CCGA019      43     A
MCG                BX7    -X2*X4      [DI] = [DI] AND ^PRS                     CCGA019      44     A
MCG                SA7    A4                                                   CCGA019      45     A
MCG                                                                            CCGA019      46     A
MCG      *         GENERATE XMT OF LOADED TEMP AT END OF TEXT                  CCGA019      47     A
MCG                                                                            CCGA019      48     A
MCG                ALLOC  TXT,8                                                CCGA019      49     A
MCG                ALLOC  TREE,3      (B6) =  LTR = OLD(L.TREE)                CCGA019      50     A
MCG                RJ     AIP                                                  CCGA019      51     A
MCG                SA1    R0                                                   CCGA019      52     A
MCG                SA2    L.TXT                                                CCGA019      53     A
MCG                SX2    X2-8        LT = L.TXT - 8                           CCGA019      54     A
MCG                SX7    B1                                                   CCGA019      55     A
MCG                SX6    B6                                                   CCGA019      56     A
MCG                LX6    I.INDXP                                              CCGA019      57     A
MCG                BX6    X6+X7                                                CCGA019      58     A
MCG                SA6    "TB"+X2     [LIT] = LW(O,O,LTR,0,1)                  CCGA019      59     A
MCG                SB3    OC.XMT                                               CCGA019      60     A
MCG                SA4    F.RDT+B3                                             CCGA019      61     A
MCG                LX7    D.PRSP                                               CCGA019      62     A
MCG                BX6    X4+X7                                                CCGA019      63     A
MCG                SA6    A6-B1       DIT=LIT-1;[DIT]=RDT(OC.XMT)!PRS          CCGA019      64     A
MCG                MX7    0                                                    CCGA019      65     A
MCG                SA7    A6-B1       R2T=DIT-1;[R2T]=0                        CCGA019      66     A
MCG                SX1    X1                                                   CCGA019      67     A
MCG                LX1    R1.RJP                                               CCGA019      68     A
MCG                BX7    X1+X2                                                CCGA019      69     A
MCG                PX6    B3,X7                                                CCGA019      70     A
MCG                SA6    A7-B1       R1T=R2T-1;[R1T]=TYI(OC.XMT,R0,0,LT)      CCGA019      71     A
MCG                LX1    -R1.RJP                                              CCGA019      72     A
MCG                                                                            CCGA019      73     A
MCG      *         GENERATE RS AFTER XMT.                                      CCGA019      74     A
MCG                                                                            CCGA019      75     A
MCG                SB2    X2+4        RSLT = LT + 4                            CCGA019      76     A
MCG                SB3    "TB"+B1                                              CCGA019      77     A
MCG                SX6    B6+2                                                 CCGA019      78     A
MCG                SX7    B1                                                   CCGA019      79     A
MCG                LX6    I.INDXP                                              CCGA019      80     A
MCG                BX7    X6+X7                                                CCGA019      81     A
MCG                SA7    "TB"+B2     [LIT] = LW(0,0,LTR+2,0,1)                CCGA019      82     A
MCG                SA3    F.RDT+OC.RS                                          CCGA019      83     A
MCG                BX7    X3                                                   CCGA019      84     A
MCG                SA7    A7-B1       DIT=LIT-1;[LIT]=F.RDT(OC.RS)             CCGA019      85     A
MCG                MX6    0                                                    CCGA019      86     A
MCG                SA6    A7-B1       R2T=DIT-1;[R2T]=0                        CCGA019      87     A
MCG                SA3    B3+X1       R1W OF RS OF R0                          CCGA019      88     A
MCG                BX6    X0*X3       MASK OUT RI                              CCGA019      89     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                BX7    X6+X2       REPLACE IT WITH LT                          CCGA019     90    A
MCG                SA7    A6-B1       R1T=R2T-1;[R1T]=TYII(OC.RS,IN,SO,LT)        CCGA019     91    A
MCG                                                                               CCGA019     92    A
MCG      *         ADJUST *SIT* SO THAT XMT IS THE LAST SUCCESSOR OF R0.          CCGA019     93    A
MCG                                                                               CCGA019     94    A
MCG                SA3    "TB"+X1                                                 CCGA019     95    A
MCG                SA4    O.TREE                                                  CCGA019     96    A
MCG                LX3    -I.INDXP                                                CCGA019     97    A
MCG                MX7    -I.INDXL                                                CCGA019     98    A
MCG                SB2    X4          =UO                                         CCGA019     99    A
MCG                BX3    -X7*X3                                                  CCGA019    100    A
MCG                SA3    B2+X3       = UI                                        CCGA019    101    A
MCG                                                                               CCGA019    102    A
MCG      *         SEARCH FOR LAST SUCCESSOR OF R0                                CCGA019    103    A
MCG                                                                               CCGA019    104    A
MCG       UJR20    LX3    59-U.LASTP                                              CCGA019    105    A
MCG                MI     X3,UJR30    IF LAST                                     CCGA019    106    A
MCG                SA3    A3+B1                                                   CCGA019    107    A
MCG                EQ     UJR20       LOOP UNTIL LAST IS FOUND                    CCGA019    108    A
MCG                                                                               CCGA019    109    A
MCG      *         MAKE LT THE LAST SUCCESSOR OF RO.                              CCGA019    110    A
MCG                                                                               CCGA019    111    A
MCG       UJR30    LX3    U.LASTP-U.SUCCP+1                                       CCGA019    112    A
MCG                BX7    X0*X3       MASK OUT SUCC.                              CCGA019    113    A
MCG                BX6    X2+X7                                                   CCGA019    114    A
MCG                LX6    U.SUCCP                                                 CCGA019    115    A
MCG                SA6    A3          LAST SUCC OF R0 = LT                        CCGA019    116    A
MCG                                                                               CCGA019    117    A
MCG      *         MAKE RS OF LT THE FIRST SUCCESSOR OF LT.                       CCGA019    118    A
MCG                                                                               CCGA019    119    A
MCG                SX4    X2+4        RSLT = LT + 4                               CCGA019    120    A
MCG                LX2    U.PREDP                                                 CCGA019    121    A
MCG                LX4    U.SUCCP                                                 CCGA019    122    A
MCG                BX7    X2+X4                                                   CCGA019    123    A
MCG                SA7    B2+B6       [LTR] = USES(0,LT,0,RSLT)                   CCGA019    124    A
MCG                                                                               CCGA019    125    A
MCG      *         MAKE THE PREVIOUS LAST SUCCESSOR OF RO THE LAST SUCCESSOR OF L CCGA019    126    A
MCG                                                                               CCGA019    127    A
MCG                BX6    "RN"X3      LSR0 = PREV. LAST SUCCESSOR OF R0           CCGA019    128    A
MCG                SX7    B1                                                      CCGA019    129    A
MCG                LX7    U.LASTP                                                 CCGA019    130    A
MCG                BX6    X6+X7                                                   CCGA019    131    A
MCG                BX7    X2+X6                                                   CCGA019    132    A
MCG                SA7    A7+B1       [LTR+1] = USES(1,LT,0,LSR0)                 CCGA019    133    A
MCG                                                                               CCGA019    134    A
MCG      *         MAKE THE PREV. LAST SUCCESSOR OF RO THE LAST SUCCESSOR OF RSLT CCGA019    135    A
MCG                                                                               CCGA019    136    A
MCG                LX4    U.PREDP-U.SUCCP                                         CCGA019    137    A
MCG                BX7    X4+X6                                                   CCGA019    138    A
MCG                SA7    A7+B1       [LTR+2] = USES(1,RSLT,0,LSR0)               CCGA019    139    A
MCG                EQ     UJR0        TRY AGAIN                                   CCGA019    140    A
MCG                                                                               CCG0008     16    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG       UJR40    SA2    T0                                                    CCG0008       17     A
MCG                SB2    "TB"-B1                                               CCG0008       18     A
MCG                S"DI"  B2+X2                                                 CCG0008       19     A
MCG                LX4    59-D.PRSP                                             CCG0008       20     A
MCG                PL     X4,UJR.E1   IF ^PRS[DI(T0)]                           CCG0008       21     A
MCG                SA3    "DI"+2      R1RS = DI + 2                             CCG0008       22     A
MCG                LX3    -R1.SOP                                               CCG0008       23     A
MCG                MX7    -SO.RTL-SO.RNL                                        CCG0008       24     A
MCG                BX6    -X7*X3                                                CCG0008       25     A
MCG                SB4    X6          REGF = SOREG[R1RS]                        CCG0008       26     A
MCG                SA2    UJRA                                                  CCG0008       27     A
MCG                SB3    X2          REGR = UJRA  */ THE REGISTER WE FREED     CCG0008       28     A
MCG                SB2    OC.XMT                                                CCG0008       29     A
MCG                RJ     OXI         OUTPUT AN XMT                             CCG0008       30     A
MCG                SA2    R0                                                    CCG0008       31     A
MCG                RJ     ISI         RELOAD THE SAVED REGISTER                 CCG0008       32     A
MCG                SA2    T0                                                    CCG0008       33     A
MCG                RJ     ISI         ISSUE T0                                  CCG0008       34     A
MCG                EQ     UJR                                                   CCG0008       35     A
MCG                                                                             CCG0008       36     A
MCG       UJRA     BSS    1                                                     CCG0008       37     A
MCG                                                                             MCG         3059     A
MCG      *         COMPILER ERROR, UNFORSEEN SITUATION                           MCG         3060     A
MCG                                                                             MCG         3061     A
MCG       DRT0     BSS    0                                                     MCG         3062     A
MCG       UJR.E1   BSS    0                                                     MCG         3063     A
MCG                PRNT   JAM,RUI                                               MCG         3064     A
MCG      *IF       DEF,TESTFTN,1                                                MCG         3065          I
         -FCC1656X
MCG                SNAPRL JAM-ERR                                              MCG         3066          I
         -FCC0641
MCG                SA1    =3LJAM                                               MCG         3067          I
         -FCC0641
MCG                CALL   PUNT        *** TEMP ***                             MCG         3068          I
         -FCC0641
MCG                EQ     *+1S17                                               FCC0641       73     A
MCG       IIX      SPACE  3,14                                                 CCG0110       18     A
MCG      **        IIX - ISSUE INSTRUCTION AND TRANSMITS                        CCG0110       19     A
MCG      *         ENTRY INSTRUCTION IN T0 IS TYPE I AND *PRS* AND REGS CLOGGED CCG0110       20     A
MCG      *         EXIT  (X3) = 0 IF ISSUED                                     CCG0110       21     A
MCG                                                                             CCG0110       22     A
MCG       IIX      ROUTINE                                                     CCG0110       23     A
MCG                SA1    "DI"-2      R1 = DI - 2                               CCG0110       24     A
MCG                LX1    -R1.RJP                                               CCG0110       25     A
MCG                RJ     COA                                                   CCG0110       26     A
MCG                NZ     X3,IIX      IF NOT AVAIL                              CCG0110       27     A
MCG                LX1    R1.RJP-R1.RKP                                         CCG0110       28     A
MCG                RJ     COA                                                   CCG0110       29     A
MCG                NZ     X3,IIX                                                CCG0110       30     A
MCG                MX5    1                                                     CCG0110       31     A
MCG                BX6    -X5*X4      PRS[DI] = 0                               CCG0110       32     A
MCG                LX6    1+D.PRSP                                              CCG0110       33     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SA6    A4                                                  CCG0110      34    A
MCG                SA2    T0                                                  CCG0110      35    A
MCG                RJ     ISI          ISSUE TO                               CCG0110      36    A
MCG                SA2    T0                                                  CCG0110      37    A
MCG                SA5    "TB"+X2      LI = TB + T0                           CCG0110      38    A
MCG                SA3    A5+B1        R1RS = LI + 1                          CCG0110      39    A
MCG                LX3    -R1.SOP                                             CCG0110      40    A
MCG                MX6    -SO.REGL                                            CCG0110      41    A
MCG                BX7    -X6*X3                                              CCG0110      42    A
MCG                SA7    IIXA                                                CCG0110      43    A
MCG                SB4    X7           SOURCE = SOREG[R1RS]                   CCG0110      44    A
MCG                SA1    AXR                                                 CCG0110      45    A
MCG                SX6    FX.SCR+FX.XT                                        CCG0110      46    A
MCG                BX7    X1*X6                                               CCG0110      47    A
MCG                NX6    B3,X7                                               CCG0110      48    A
MCG                SA2    B3+FXRA+16-47                                       CCG0110      49    A
MCG                SB3    X2           RESULT = FXR(SCR)                      CCG0110      50    A
MCG                SB2    OC.XMT                                              CCG0110      51    A
MCG                RJ     OXI          MOVE SOURCE TO A FREE REG              CCG0110      52    A
MCG                SA1    IIXA                                                CCG0110      53    A
MCG                SA2    T0                                                  CCG0110      54    A
MCG                SA3    "TB"+X2                                             CCG0110      55    A
MCG                SB4    X3           SOURCE = REG[LI]                       CCG0110      56    A
MCG                SB3    X1           RESULT = SOREG[R1RS]                   CCG0110      57    A
MCG                SB2    OC.XMT                                              CCG0110      58    A
MCG                RJ     OXI          MOVE T0 TO SOREG[R1RS]                 CCG0110      59    A
MCG                MX3    0                                                   CCG0110      60    A
MCG                EQ     IIX                                                 CCG0110      61    A
MCG                                                                           CCG0110      62    A
MCG        IIXA    BSS    1                                                   CCG0110      63    A
MCG        COA     SPACE  3,14                                                CCG0110      64    A
MCG        COA     ROUTINE                                                    CCG0110      65    A
MCG                BX6    "RN"X1                                              CCG0110      66    A
MCG                SA2    "TB"+X6      LI = TB + RI[R1]                       CCG0110      67    A
MCG                SA3    RVT+X2                                              CCG0110      68    A
MCG                MI     X3,COA       IF REG LOCKED                          CCG0110      69    A
MCG                SA2    A2-B1        DI = LI - 1                            CCG0110      70    A
MCG                MX3    -D.USESL                                            CCG0110      71    A
MCG                LX3    -D.USESP                                            CCG0110      72    A
MCG                BX7    -X3*X2                                              CCG0110      73    A
MCG                SX3    X7-1         X3 = 0 IF ONLY 1 USE                   CCG0110      74    A
MCG                EQ     COA                                                 CCG0110      75    A
MCG        FSR     SPACE  3,14                                                CCG19         3    A
MCG      **        FSR - FIND STORE REGISTER, NO REGS AVAIL                   CCG19         4    A
MCG      *                                                                    CCG19         5    A
MCG      *         EXIT   (B3) = REGNO, = 0 IF NOTHING FOUND                  CCG19         6    A
MCG      *                (B4) = MCOST                                        CCG19         7    A
MCG      *         (X5) = NEG IF BOTH X6 AND X7 LOCKED.                       CCGA019     141    A
MCG                                                                           CCG19         8    A
MCG        FSR     ROUTINE                                                    CCG19         9    A
MCG                SA1    RVT+RN.X0+6                                         CCG19        10    A
MCG                SA2    A1+B1                                               CCG19        11    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                MX5    1                                                    CCG19        12     A
MCG                BX6    X5*X1                                                CCG19        13     A
MCG                BX7    X5*X2                                                CCG19        14     A
MCG                SA6    FSRA        FSRA(1) = LOCK[RVT(X6)]                  CCG19        15     A
MCG                SA7    A6+B1       FSRA(2) = LOCK[RVT(X7)]                  CCG19        16     A
MCG                SA3    CLASS                                                CCG19        17     A
MCG                SX4    X3-FX.SCR                                            CCG19        18     A
MCG                ZR     X4,FSR2     IF CLASS = FX.SCR                        CCG19        19     A
MCG                SA2    T0                                                   CCG19        20     A
MCG                RJ     GOR         GET REGNO"S OF OPERANDS OF T0            CCG19        21     A
MCG                                                                            CCG19        22     A
MCG       FSR1     ZR     X7,FSR2     IF END OF LIST                           CCG19        23     A
MCG                SX4    X7-RN.X0-6                                           CCG19        24     A
MCG                AX7    18                                                   CCG19        25     A
MCG                MI     X4,FSR1     IF NOT A STORE REG                       CCG19        26     A
MCG                MX6    1                                                    CCG19        27     A
MCG                SA6    FSRA+X4     INDICATE REGISTER IS NOT AVAIL           CCG19        28     A
MCG                EQ     FSR1                                                 CCG19        29     A
MCG                                                                            CCG19        30     A
MCG       FSR2     SA1    FSRA                                                 CCG19        31     A
MCG                SA2    A1+B1                                                CCG19        32     A
MCG                BX6    X1*X2                                                CCG19        33          I
          -CCGA019
MCG                SB3    B0          REGNO = 0                                CCG19        34          I
          -CCGA019
MCG                BX5    X1*X2                                                CCGA019     142     A
MCG                SB3    RN.X0+6     REGNO = X6                               CCGA019     143     A
MCG                SB4    B1+B1       MCOST = 2                                CCG19        35     A
MCG                MI     X6,FSR      IF BOTH X6 AND X7 UNAVAILABLE            CCG19        36          I
          -CCGA019
MCG                SB3    RN.X0+6     REGNO = X6                               CCG19        37          I
          -CCGA019
MCG                MI     X5,FSR3     IF BOTH X6 AND X7 LOCKED                 CCGA019     144     A
MCG                BX7    X1+X2                                                CCG19        38     A
MCG                ZR     X7,FSR3     IF BOTH AVAILABLE                        CCG19        39     A
MCG                ZR     X1,FSR      IF X6 IS AVAILABLE                       CCG19        40     A
MCG                SB3    B3+B1       REGNO = RN.X7                            CCG19        41     A
MCG                EQ     FSR                                                  CCG19        42     A
MCG                                                                            CCG19        43     A
MCG       *        BOTH ST REGS AVAIL, SEARCH FOR NEXT USE OF EACH.             CCG19        44     A
MCG                                                                            CCG19        45     A
MCG       FSR3     SA3    O.TREE                                               CCG19        46     A
MCG                SA1    RVT+RN.X0+6                                          CCG19        47     A
MCG                SA2    A1+B1                                                CCG19        48     A
MCG                MX7    -I.INDXL                                             CCG19        49     A
MCG                SB2    X3-1                                                 CCG19        50     A
MCG                SA1    "TB"+X1     LX6 = TB + RVT(X6)                       CCG19        51     A
MCG                SA2    "TB"+X2     LX7 = TB + RVT(X7)                       CCG19        52     A
MCG                LX1    -I.INDXP                                             CCG19        53     A
MCG                LX2    -I.INDXP                                             CCG19        54     A
MCG                BX1    -X7*X1                                               CCG19        55     A
MCG                BX2    -X7*X2                                               CCG19        56     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SB6    59-U.EQVP                                          CCG19        57     A
MCG                SA1    B2+X1       UX6 = TREE-1 + INDX[LX6]               CCG19        58     A
MCG                SA2    B2+X2       UX7 = TREE-1 + INDX[LX7]               CCG19        59     A
MCG                                                                          CCG19        60     A
MCG       FSR4     SA1    A1+B1       UX6 = UX6 + 1                          CCG19        61     A
MCG                LX6    B6,X1                                              CCG19        62     A
MCG                MI     X6,FSR4     IF EQV[UX6]                            CCG19        63     A
MCG                LX1    -U.SUCCP                                           CCG19        64     A
MCG                SA4    "TB"+X1     LIS = TB + SUCC[UX6]                   CCG19        65     A
MCG                LX4    59-I.IIP                                           CCG19        66     A
MCG                MI     X4,FSR4     IF II[LIS]                             CCG19        67     A
MCG                                                                          CCG19        68     A
MCG       FSR5     SA2    A2+B1       UX7 = UX7 + 1                          CCG19        69     A
MCG                LX6    B6,X2                                              CCG19        70     A
MCG                MI     X6,FSR5     IF EQV[UX7]                            CCG19        71     A
MCG                LX1    -U.SUCCP                                           CCG19        72     A
MCG                SA4    "TB"+X2     LIST = TB + SUCC[UX7]                  CCG19        73     A
MCG                LX4    59-I.IIP                                           CCG19        74     A
MCG                MI     X4,FSR5     IF II[LIS]                             CCG19        75     A
MCG                                                                          CCG19        76     A
MCG                SB5    X1                                                 CCG19        77     A
MCG                SB6    X2                                                 CCG19        78     A
MCG                GT     B5,B6,FSR   REGNO = IF SUCC[UX6]  > SUCC[UX7]      CCG19        79          I
          -CCGA023
MCG                SB3    B3+1              THEN RN.X6; ELSE RN.X7           CCG19        80          I
          -CCGA023
MCG                GT     B5,B6,FSR   IF ( SUCC[UI6] > SUCC[UI7] )           CCGA023      22     A
MCG                SB3    B3+B1             THEN RN.X6   ELSE RN.X7          CCGA023      23     A
MCG                NE     B5,B6,FSR   IF SUCC[UX6] " SUCC[UX7]               CCGA023      24     A
MCG                SA2    RVT+B3      RX7 = RVT + RN.X7                      CCGA023      25     A
MCG                SA1    A2-B1       RX6 = RX7 - 1                          CCGA023      26     A
MCG                MX7    -I.USESL                                           CCGA023      27     A
MCG                SA1    "TB"+X1     LIX6 = BASE + RX6                      CCGA023      28     A
MCG                SA2    "TB"+X2     LIX7 = BASE + RX7                      CCGA023      29     A
MCG                LX1    -I.USESP                                           CCGA023      30     A
MCG                LX2    -I.USESP                                           CCGA023      31     A
MCG                BX1    -X7*X1                                             CCGA023      32     A
MCG                BX2    -X7*X2                                             CCGA023      33     A
MCG                SB5    X1                                                 CCGA023      34     A
MCG                SB6    X2                                                 CCGA023      35     A
MCG                GT     B6,B5,FSR    IF USES[LIX7] > USES[LIX6]            CCGA023      36     A
MCG                SB3    B3-B1       REGNO = RN.X6                          CCGA023      37     A
MCG                EQ     FSR                                                CCG19        81     A
MCG                                                                          CCG19        82     A
MCG       FSRA     BSS    2                                                  CCG19        83     A
MCG       GOR      SPACE  3,14                                               CCG19        84     A
MCG      **        GOR - GET OPERAND REGISTERS OF INSTRUCTIONS THAT ARE      CCG19        85     A
MCG      *                A) NOT LOCKED                                     CCG19        86     A
MCG      *                B) NOT IN *CLASS*                                 CCG19        87     A
MCG      *                                                                  CCG19        88     A
MCG      *         ENTRY  (X2) = R-NUMBER OF INSTRUCTION TO CHECK OPERANDS OF CCG19        89     A
MCG      *                                                                  CCG19        90     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG       *         EXIT    (X7) = LOR, LIST OF REGISTERS                    CCG19        91     A
MCG                                                                          CCG19        92     A
MCG        GOR      ROUTINE                                                  CCG19        93     A
MCG                 SB6     "TB"-B1                                          CCG19        94     A
MCG                 S"DI"   B6+X2        DI = BASE-1 + T0                    CCG19        95     A
MCG                 LX4     58-D.TYP                                         CCG19        96     A
MCG                 MX7     0            LOR = 0                             CCG19        97     A
MCG                 IX6     X4+X4                                            CCG19        98     A
MCG                 MI      X6,GOR       IF TYPE[DI] = II ! IV               CCG19        99     A
MCG                 PL      X4,GOR1      IF TYPE[DI] = I                     CCG19       100     A
MCG                                                                          CCG19       101     A
MCG       *         ASSEMBLE OPERAND R-NUMBERS                               CCG19       102     A
MCG                                                                          CCG19       103     A
MCG                 SA3     "DI"-B1      R2 = DI - 1                         CCG19       104     A
MCG                 LX3     -IH.RFP                                          CCG19       105     A
MCG                 BX5     "RN"X3       RNUMS = RF[R2]                      CCG19       106     A
MCG                 EQ      GOR2                                             CCG19       107     A
MCG                                                                          CCG19       108     A
MCG        GOR1     LX4     2+D.TYP-D.TYIOP                                  CCG19       109     A
MCG                 MX6     -D.TYIOL                                         CCG19       110     A
MCG                 SA5     "DI"-2       R1 = DI - 2                         CCG19       111     A
MCG                 BX7     -X6*X4                                           CCG19       112     A
MCG                 SA3     TYIOM+X7                                         CCG19       113     A
MCG                 UX6     B2,X3                                            CCG19       114     A
MCG                 BX2     X6*X5                                            CCG19       115     A
MCG                 AX5     B2,X2        RNUMS = TYIO[R1,DI]                 CCG19       116     A
MCG                                                                          CCG19       117     A
MCG        GOR2     MX4     1                                                CCG19       118     A
MCG                 LX4     1+RV.LOCKP                                       CCG19       119     A
MCG                 SA1     CLASS                                            CCG19       120     A
MCG                                                                          CCG19       121     A
MCG        GOR3     BX6     "RN"X5       R = RI[RNUMS]                       CCG19       122     A
MCG                 AX5     R1.RIL       RNUMS = SHIFT( RNUMS , -16 )        CCG19       123     A
MCG                 ZR      X6,GOR4      IF R = 0                            CCG19       124     A
MCG                 SA3     "TB"+X6      L = BASE + R                        CCG19       125     A
MCG                 LX3     -I.REGP      REGNO = REG[L]                      CCG19       126     A
MCG                 SA2     RBV+X3                                           CCG19       127     A
MCG                 BX6     X2*X1                                            CCG19       128     A
MCG                 NZ      X6,GOR4      IF RBV(REGNO) & CLASS " 0  */ REG IN CLASS  CCG19  129  A
MCG                 SA2     RVT+X3                                           CCG19       130     A
MCG                 MI      X2,GOR4      IF LOCK[RVT(REGNO)]                 CCG19       131     A
MCG                 LX7     18                                               CCG19       132     A
MCG                 SX3     X3                                               CCG19       133     A
MCG                 BX7     X3+X7        LOR = SHIFT(LOR,18) ! REGNO         CCG19       134     A
MCG        GOR4     NZ      X5,GOR3      IF RNUMS " 0                        CCG19       135     A
MCG                                                                          CCG19       136     A
MCG                 EQ      GOR                                              CCG19       137     A
MCG        RII      TITLE   RII - RESET ISSUED INSTRUCTION                   MCG        3069     A
MCG       **        RII - RESET ISSUED INSTRUCTION FOR REISSUE               MCG        3070     A
MCG       *                                                                  MCG        3071     A
MCG       *         ENTRY   (X1) = R , *TXT* INDEX OF INSTRUCTION            MCG        3072     A
MCG                                                                          MCG        3073     A
```

```
          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG       RII      ROUTINE                                                    MCG        3074     A
MCG                RJ     RIL         RESET ISSUE LIST FOR R                  MCG        3075     A
MCG                SB6    "TB"-B1                                             MCG        3076     A
MCG                S"DI"  B6+B3       DI = BASE-1 + R0                        MCG        3077     A
MCG                LX4    59-D.ZPP                                            MCG        3078     A
MCG                MI     X4,RII      IF ZP[DI]     */ NOT A *LD* WITH OPERANDS MCG        3079     A
MCG                                                                           MCG        3080     A
MCG      *         ADJUST USES COUNTS OF OPERAND REGS, SET FOR REISSUE       MCG        3081     A
MCG                                                                           MCG        3082     A
MCG                LX4    1+D.ZPP+58-D.TYP                                    MCG        3083     A
MCG                PL     X4,RII1     IF TYPE[DI] = I                         MCG        3084     A
MCG                SA3    "DI"-B1     R2 = DI - 1                             MCG        3085     A
MCG                LX3    -IH.RFP                                             MCG        3086     A
MCG                BX5    "RN"X3      RNUMS = RF[R2]                          MCG        3087     A
MCG                EQ     RII2                                                MCG        3088     A
MCG                                                                           MCG        3089     A
MCG       RII1     SA3    "DI"-2      R1 = DI - 2                             MCG        3090     A
MCG                SA2    TYIOM                                               MCG        3091     A
MCG                UX6    B2,X2                                               MCG        3092     A
MCG                BX7    X6*X3                                               MCG        3093     A
MCG                AX5    B2,X7       RNUMS = TYIO[R1,0]                      MCG        3094     A
MCG                                                                           MCG        3095     A
MCG       RII2     SX4    B1                                                  MCG        3096     A
MCG                LX4    I.USESP                                             MCG        3097     A
MCG                                                                           MCG        3098     A
MCG       RII3     BX6    "RN"X5      R = RI[RNUMS]                           MCG        3099     A
MCG                AX5    R1.RIL      RNUMS = SHIFT( RNUMS , -16 )            MCG        3100     A
MCG                ZR     X6,RII4     IF R = 0                                MCG        3101     A
MCG                SA3    "TB"+X6     L = BASE + R                            MCG        3102     A
MCG                IX7    X3+X4       USES[L] = USES[L] + 1                   MCG        3103     A
MCG                SA7    A3                                                  MCG        3104     A
MCG                LX3    -I.REGP     REGNO = REG[L]                          MCG        3105     A
MCG                SB3    X3                                                  MCG        3106     A
MCG                SA2    RVT+X3                                              MCG        3107     A
MCG                BX1    X2-X6                                               MCG        3108     A
MCG                BX3    "RN"X1                                              MCG        3109     A
MCG                ZR     X3,RII4     IF RI[RVT(REGNO)] = R                   MCG        3110     A
MCG                SA1    AXR                                                 MCG        3111     A
MCG                SA3    RBV+B3                                              MCG        3112     A
MCG                SA6    A2          RVT(REGNO) = R                          MCG        3113     A
MCG                BX7    -X3*X1      AXR = AXR & ^RBV(REGNO)                 MCG        3114     A
MCG                SA7    A1                                                  MCG        3115     A
MCG       RII4     NZ     X5,RII3     IF RNUMS " 0                            MCG        3116     A
MCG                                                                           MCG        3117     A
MCG                EQ     RII                                                 MCG        3118     A
MCG       CSI      TITLE  CSI - CHECK STORE ISSUE                             MCG        3119     A
MCG      **        CSI - CHECK STORE REGS FOR ISSUE OF A STORE INSTRUCTION   MCG        3120     A
MCG      *                RECOMPUTE PS BIT FOR RESULTS IN STORE REGS        MCG        3121     A
MCG                                                                           MCG        3122     A
MCG       CSI      ROUTINE                                                    MCG        3123     A
MCG                SB3    RN.X0+6     I = RN.X6                               MCG        3124     A
MCG                                                                           MCG        3125     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG       CSI1     SA1    RVT+B3      R = RVT(I)                                  MCG        3126     A
MCG                SB6    "TB"-B1                                                 MCG        3127     A
MCG                SA4    B6+X1       DI = BASE-1 + R                             MCG        3128     A
MCG                LX4    59-D.PSP                                                MCG        3129     A
MCG                PL     X4,CSI4     IF ^PS[DI]    */ NOT A STORE PREDECESSOR    MCG        3130     A
MCG                                                                               MCG        3131     A
MCG       *        CHECK FOR ISSUABLE STORES, ADJUST PS BIT                        MCG        3132     A
MCG                                                                               MCG        3133     A
MCG                SA5    "DI"+B1     L = DI + 1                                  MCG        3134     A
MCG                MX7    -I.INDXL    ^P = 1                                      MCG        3135     A
MCG                SA2    O.TREE                                                  MCG        3136     A
MCG                LX5    -I.INDXP                                                MCG        3137     A
MCG                BX6    -X7*X5                                                  MCG        3138     A
MCG                SB2    X2                                                      MCG        3139     A
MCG                MX2    1                                                       MCG        3140     A
MCG                SA3    B2+X6       UI = O.TREE + INDX[L]                       MCG        3141     A
MCG                                                                               MCG        3142     A
MCG       CSI2     LX3    -U.SUCCP                                                MCG        3143     A
MCG                SA5    "TB"+X3     SI = BASE + SUCC[UI]                        MCG        3144     A
MCG                LX5    59-I.IIP                                                MCG        3145     A
MCG                MI     X5,CSI3     IF II[SI]                                   MCG        3146     A
MCG                SA1    A5-B1       D = SI - 1                                  MCG        3147     A
MCG                LX5    I.IIP-I.ICP                                             MCG        3148     A
MCG                LX1    59-D.STP                                                MCG        3149     A
MCG                BX6    X1*X5                                                   MCG        3150     A
MCG                MI     X6,CSI5     IF ST[D] & IC[SI]                           MCG        3151     A
MCG       #DAL     IFNE   .DAL,0                                                  MCG        3152     A
MCG                BX5    X2*X1                                                   MCG        3153     A
MCG                LX1    D.STP-D.L2P                                             MCG        3154     A
MCG                BX1    -X1*X5                                                  MCG        3155     A
MCG       #DAL     ENDIF                                                          MCG        3156     A
MCG                BX7    -X1*X7      ^P = ^P & ^(ST[D] &^L2[D])                  MCG        3157     A
MCG       CSI3     LX3    U.SUCCP+59-U.LASTP                                      MCG        3158     A
MCG                BX6    X3          UL = UI                                     MCG        3159     A
MCG                SA3    A3+B1       UI = UI + 1                                 MCG        3160     A
MCG                PL     X6,CSI2     IF ^LAST[UL]                                MCG        3161     A
MCG                                                                               MCG        3162     A
MCG                PL     X7,CSI4     IF ^P = 0     */ RESULT STILL IS *PS*       MCG        3163     A
MCG                BX7    -X2*X4      PS[DI] = 0                                  MCG        3164     A
MCG                LX7    1+D.PSP                                                 MCG        3165     A
MCG                SA7    A4                                                      MCG        3166     A
MCG                                                                               MCG        3167     A
MCG       CSI4     SB3    B3+B1       I = I + 1                                   MCG        3168     A
MCG                SX6    B3-RN.X0-7                                              MCG        3169     A
MCG                ZR     X6,CSI1     IF I = RN.X7                                MCG        3170     A
MCG                EQ     CSI                                                     MCG        3171     A
MCG                                                                               MCG        3172     A
MCG       CSI5     SX2    A5-"TB"     R = SI - BASE                               MCG        3173     A
MCG                RJ     ISI         ISSUE THE STORE                             MCG        3174     A
MCG                SB3    B3-AMX      REGNO = REGNO - AMX */ RESTORE REGNO ( *ASC FCC1555X      19          I
          -FCC2038
MCG                SB3    RN.X0+6     I = RN.X6     */ RESET                      FCC2038         5     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                EQ     CSI1                                                    MCG        3175     A
MCG        DRT     TITLE  DRT - DETERMINE RESULT REGISTER TYPE ( CLASS )          MCG        3176     A
MCG       **       DRT - DETERMINE RESULT REGISTER TYPE ( CLASS ) NEEDED          MCG        3177     A
MCG       *                                                                       MCG        3178     A
MCG       *        ENTRY  (A4,X4) = DI , [DI]                                     MCG        3179     A
MCG       *                                                                       MCG        3180     A
MCG       *        EXIT   (X6) = REGISTER CLASS BITS                              MCG        3181     A
MCG                                                                               MCG        3182     A
MCG        DRT     ROUTINE                                                        MCG        3183     A
MCG                LX4    59-D.NRP                                                MCG        3184     A
MCG                SX6    FX.SCR      CLASS = FX.SCR                              MCG        3185     A
MCG                NX7    B2,X4                                                   MCG        3186     A
MCG                JP     DRT1+B2     JUMP( NORMC[DI] )                           MCG        3187     A
MCG                                                                               MCG        3188     A
MCG        DRT1    BSS    0                                                       MCG        3189     A
MCG                SA5    "DI"+2      R1RS = DI + 2                               MCG        3190     A
MCG                EQ     DRT2                                                    MCG        3191     A
MCG                                                                               MCG        3192     A
MCG                SX6    FX.ST       CLASS = FX.ST                               MCG        3193     A
MCG                EQ     DRT                                                     MCG        3194     A
MCG                                                                               MCG        3195     A
MCG                SA5    "DI"+2      R1RS = DI + 2                               MCG        3196     A
MCG                EQ     DRT3                                                    MCG        3197     A
MCG                                                                               MCG        3198     A
MCG                EQ     DRT0        ECJ                                         MCG        3199     A
MCG                                                                               MCG        3200     A
MCG                EQ     DRT         LVL2                                        MCG        3201     A
MCG                                                                               MCG        3202     A
MCG                SX6    FX.LD       CLASS = LD                                  MCG        3203     A
MCG                EQ     DRT                                                     MCG        3204     A
MCG                                                                               MCG        3205     A
MCG                EQ     DRT0        ST                                          MCG        3206     A
MCG                EQ     DRT0        JP                                          MCG        3207     A
MCG                EQ     DRT         SXT                                         MCG        3208     A
MCG                EQ     DRT0        KS                                          MCG        3209     A
MCG                EQ     DRT0        PI                                          MCG        3210     A
MCG                EQ     DRT         MUC                                         MCG        3211     A
MCG                EQ     DRT         SCR                                         MCG        3212     A
MCG                                                                               MCG        3213     A
MCG       *        INSTRUCTION PRECEEDS AN *RJRS*, ENSURE B-REG IS AVAILABLE      MCG        3214     A
MCG                                                                               MCG        3215     A
MCG        DRT2    LX5    -R1.SOP                                                 MCG        3216          I
          -FCC0641
MCG                MX7    -SO.REGL                                                MCG        3217          I
          -FCC0641
MCG                BX3    -X7*X5                                                  MCG        3218          I
          -FCC0641
MCG                LX4    D.NRP-D.PSP                                             MCG        3219          I
          -FCC0641
MCG                SA2    RVT+X3                                                  MCG        3220          I
          -FCC0641
MCG                SA1    "TB"+X2     LP = TB + RVT(REGNO)                        MCG        3221          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

          -FCC0641
MCG                 MX7    -I.USESL                                          MCG        3222          I
          -FCC0641
MCG                 LX7    I.USESP                                           MCG        3223          I
          -FCC0641
MCG                 BX7    -X7*X1                                            MCG        3224          I
          -FCC0641
MCG                 NZ     X7,DRT0     IF USES[LP] " 0  */ REG IS IN USE     MCG        3225          I
          -FCC0641
MCG        DRT2     RJ     FBR         REGNO = FBR( R1RS )                   FCC0641      74     A
MCG                 MI     B3,DRT0     IF ACT < 0    */ NO AVAILABLE B-REG   FCC0641      75     A
MCG                 LX4    D.NRP-D.PSP                                       FCC0641      76     A
MCG                 SX6    FX.SCR      CLASS = FX.SCR                        FCC0641      77     A
MCG                 PL     X4,DRT      IF ^PS[DI]                            MCG        3226     A
MCG                 SX6    FX.ST       CLASS = FX.ST                         MCG        3227     A
MCG                 EQ     DRT                                               MCG        3228     A
MCG        DRT      SPACE  1,10                                              MCG        3229     A
MCG       *         INSTRUCTION PRECEEDS A *RS*                              MCG        3230     A
MCG                                                                          MCG        3231     A
MCG        DRT3     LX5    -R1.SOP                                           MCG        3232     A
MCG                 MX7    -SO.REGL                                          MCG        3233     A
MCG                 BX3    -X7*X5                                            MCG        3234     A
MCG                 SA2    RBV+X3                                            MCG        3235     A
MCG                 SX6    X2          CLASS = RBV( SOREG[R1RS] )            MCG        3236     A
MCG                 EQ     DRT                                               MCG        3237     A
MCG        FRR      TITLE  FRR - FIND RESULT REGISTER                        MCG        3238     A
MCG       **        FRR - FIND RESULT REGISTER                               MCG        3239     A
MCG       *                                                                  MCG        3240     A
MCG       *         ENTRY  (X1) = CLASS OF REGISTERS WHICH RESULT SHOULD BE IN MCG        3241     A
MCG       *                (T0) = *TXT* INDEX OF INSTRUCTION TO BE ISSUED    MCG        3242     A
MCG       *                                                                  MCG        3243     A
MCG       *         EXIT   (B3) = REGF , REGNO OF REGISTER THAT IS CHEIPEST TO FREE MCG 3244     A
MCG       *                (B4) = MCOST , COST OF RESTORING *REGF*           MCG        3245     A
MCG       *                                                                  MCG        3246     A
MCG       *         CALLS  CRC                                               MCG        3247     A
MCG                                                                          MCG        3248     A
MCG        FRR0     SB3    B0                                                MCG        3249     A
MCG                 SB6    B0          ABR = 0                               MCG        3250     A
MCG                 RJ     CRC         REGF = CRC(RVT,MCOST)                 MCG        3251     A
MCG                                                                          MCG        3252     A
MCG        FRR      ROUTINE                                                  MCG        3253     A
MCG                 SX6    X1-FX.SCR                                         MCG        3254     A
MCG                 ZR     X6,FRR0     IF CLASS = FX.SCR                     MCG        3255     A
MCG                 SA2    T0                                                MCG        3256     A
MCG                 SB6    "TB"-B1                                           MCG        3257          I
          -CCG19
MCG                 S"DI"  B6+X2       DI = BASE-1 + T0                      MCG        3258          I
          -CCG19
MCG                 LX4    58-D.TYP                                          MCG        3259          I
          -CCG19
MCG                 IX7    X4+X4                                             MCG        3260          I
          -CCG19

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                 MI     X7,FRR0     IF TYPE[DI] = II ! IV                          MCG        3261        I
          -CCG19
MCG                 PL     X4,FRR1     IF TYPE[DI] = I                                MCG        3262        I
          -CCG19
MCG                                                                                   MCG        3263        I
          -CCG19
MCG       *         ASSEMBLE OPERAND R-NUMBERS                                         MCG        3264        I
          -CCG19
MCG                                                                                   MCG        3265        I
          -CCG19
MCG                 SA3    "DI"-B1     R2 = DI - 1                                    MCG        3266        I
          -CCG19
MCG                 LX3    -IH.RFP                                                    MCG        3267        I
          -CCG19
MCG                 BX5    "RN"X3      RNUMS = RF[R2]                                 MCG        3268        I
          -CCG19
MCG                 EQ     FRR2                                                       MCG        3269        I
          -CCG19
MCG                                                                                   MCG        3270        I
          -CCG19
MCG        FRR1     LX4    2+D.TYP-D.TYIOP                                            MCG        3271        I
          -CCG19
MCG                 MX6    -D.TYIOL                                                   MCG        3272        I
          -CCG19
MCG                 SA5    "DI"-2      R1 = DI - 2                                    MCG        3273        I
          -CCG19
MCG                 BX7    -X6*X4                                                     MCG        3274        I
          -CCG19
MCG                 SA3    TYIOM+X7                                                   MCG        3275        I
          -CCG19
MCG                 UX6    B2,X3                                                      MCG        3276        I
          -CCG19
MCG                 BX2    X6*X5                                                      MCG        3277        I
          -CCG19
MCG                 AX5    B2,X2       RNUMS = TYIO[R1,DI]                            MCG        3278        I
          -CCG19
MCG                                                                                   MCG        3279        I
          -CCG19
MCG       *         LOCKOUT OPERANDS OF *T0* THAT ARE NOT IN *CLASS* SO *CRC*          MCG        3280        I
          -CCG19
MCG       *         WON"T CONSIDER THEM.                                               MCG        3281        I
          -CCG19
MCG                                                                                   MCG        3282        I
          -CCG19
MCG        FRR2     MX4    1                                                          MCG        3283        I
          -CCG19
MCG                 LX4    1+RV.LOCKP                                                 MCG        3284        I
          -CCG19
MCG                 MX7    0           I = 0         */ LIST OF LOCKED OPERAND REG    MCG        3285        I
          -CCG19
MCG                                                                                   MCG        3286        I
          -CCG19

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG       FRR3     BX6    "RN"X5      R = RI[RNUMS]                                  MCG        3287        I
          -CCG19
MCG                AX5    R1.RIL      RNUMS = SHIFT( RNUMS , -16 )                   MCG        3288        I
          -CCG19
MCG                ZR     X6,FRR4     IF R = 0                                       MCG        3289        I
          -CCG19
MCG                SA3    "TB"+X6     L = BASE + R                                   MCG        3290        I
          -CCG19
MCG                LX3    -I.REGP     REGNO = REG[L]                                 MCG        3291        I
          -CCG19
MCG                SA2    RBV+X3                                                     MCG        3292        I
          -CCG19
MCG                BX6    X2*X1                                                      MCG        3293        I
          -CCG19
MCG                NZ     X6,FRR4     IF RBV(REGNO) & CLASS " 0 */ REG IN CLASS      MCG        3294        I
          -CCG19
MCG                SA2    RVT+X3                                                     MCG        3295        I
          -CCG19
MCG                MI     X2,FRR4     IF LOCK[ RVT(REGNO) ]                          MCG        3296        I
          -CCG19
MCG                LX7    18                                                         MCG        3297        I
          -CCG19
MCG                SX3    X3                                                         MCG        3298        I
          -CCG19
MCG                BX6    X4+X2                                                      MCG        3299        I
          -CCG19
MCG                IX7    X3+X7       I = SHIFT( I , 18) ! REGNO                     MCG        3300        I
          -CCG19
MCG                SA6    A2          LOCK[ RVT(REGNO) ] = 1                         MCG        3301        I
          -CCG19
MCG       FRR4     NZ     X5,FRR3     IF RNUMS " 0                                   MCG        3302        I
          -CCG19
MCG                                                                                  MCG        3303        I
          -CCG19
MCG                SA7    FRRA        FRRA = I                                       MCG        3304        I
          -CCG19
MCG                SB3    B0                                                         MCG        3305        I
          -CCG19
MCG                RJ     GOR         GET REGNO"S OF OPERANDS OF T0                  CCG19       138     A
MCG                                                                                  CCG19       139     A
MCG       *        LOCKOUT THE OPERANDS OF *T0* THAT ARE NOT IN *CLASS* SO *CRC*     CCG19       140     A
MCG       *        WON"T CONSIDER THEM.                                              CCG19       141     A
MCG                                                                                  CCG19       142     A
MCG                SA7    FRRA        FRRA = LIST                                    CCG19       143     A
MCG                MX5    1                                                          CCG19       144     A
MCG       FRR1     SX3    X7          REGNO = SETX(I)                                CCG19       145     A
MCG                ZR     X3,FRR2     IF REGNO = 0  */ END OF LIST                   CCG19       146     A
MCG                AX7    18          I = SHIFT(I,-18)                               CCG19       147     A
MCG                SA2    RVT+X3                                                     CCG19       148     A
MCG                BX6    X2+X5       LOCK[RVT(REGNO)] = 1                           CCG19       149     A
MCG                SA6    A2                                                         CCG19       150     A
MCG                EQ     FRR1                                                       CCG19       151     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                                                                    CCG19      152    A
MCG      FRR2    SB3    B0                                             CCG19      153    A
MCG              SB6    B0          ABR = 0                            MCG       3306    A
MCG              RJ     CRC         REGF = CRC(RVT,MCOST)              MCG       3307    A
MCG                                                                    MCG       3308    A
MCG      *       UNLOCK THE LOCKED OPERAND REGISTERS                   MCG       3309    A
MCG                                                                    MCG       3310    A
MCG              SA1    FRRA        I = FRRA                           MCG       3311    A
MCG              MX4    1                                              MCG       3312    A
MCG              LX4    1+RV.LOCKP                                     MCG       3313    A
MCG                                                                    MCG       3314    A
MCG      FRR5    SA2    RVT+X1                                         MCG       3315    A
MCG              BX6    -X4*X2      LOCK[RVT(I)] = 0                   MCG       3316    A
MCG              AX1    18          I = SHIFT( I , -18 )               MCG       3317    A
MCG              SA6    A2                                             MCG       3318    A
MCG              NZ     X1,FRR5                                        MCG       3319    A
MCG              EQ     FRR                                            MCG       3320    A
MCG                                                                    MCG       3321    A
MCG      FRRA    BSS    1                                              MCG       3322    A
MCG      CRC     TITLE  CRC - COMPUTE RESTORE COSTS                    MCG       3323    A
MCG      **      CRC - COMPUTE RESTORE COSTS OF RESTABLISHING A RESULT IN A REG  MCG  3324  A
MCG      *                                                             MCG       3325    A
MCG      *       ENTRY  (B3) = REGNO, IF 0 THEN CHECK ALL X-REGS, ELSE CHECK  MCG  3326  A
MCG      *                      ONLY "REGNO".                          MCG       3327    A
MCG      *              (B6) = ABR " 0 IF B-REGISTERS AVAIL FOR A MOVE TO THEM  MCG  3328  A
MCG      *                                                             MCG       3329    A
MCG      *       EXIT   (B3) = REGNO OF REGISTER WHICH IS CHEIPEST TO FREE  MCG  3330  A
MCG      *              (B4) = RESTORATION COST                        MCG       3331    A
MCG      *                                                             MCG       3332    A
MCG      *       PRESERVES - X0, B1, B7                                MCG       3333    A
MCG      *                                                             MCG       3334    A
MCG      *       THE COST OF RESTORING A REGISTER ARE -                MCG       3335    A
MCG      *       COST   INSTRUCTION PROPERITIES / RESTORE ACTION       MCG       3336    A
MCG      *                                                             MCG       3337    A
MCG      *       -1     RF[DI] " 0 / MOVE RESULT TO AN AVAILABLE B-REGISTER  MCG  3338  A
MCG      *                                                             MCG       3339    A
MCG      *       0      ZP[DI] & ^LD[DI] / REISSUE INSTRUCTION         MCG       3340    A
MCG      *                                                             MCG       3341    A
MCG      *       1      LD[DI] & ^(L2[DI] ! FP[DI]) & NO INTERFERING STORE ISSU  MCG  3342  A
MCG      *                 IF ZP[DI] ! OPERANDS"S ARE IN B-REGS , THEN REISSUE  MCG  3343  A
MCG      *                                                             MCG       3344    A
MCG      *       2      RESULT IN X6/X7 WITH NO SPECIAL PROPERTIES     MCG       3345    A
MCG      *                 STORE THE RESULT IN A TEMP CELL AND SETUP A *LD*  MCG  3346  A
MCG      *                 OF IT ( OT.+CA ).                           MCG       3347    A
MCG                                                                    MCG       3348    A
MCG      RF.SAV  EQU    0           TURN OFF *RF* LOGIC                MCG       3349    A
MCG                                                                    MCG       3350    A
MCG      CRC0    SB6    RVT                                            MCG       3351    A
MCG              SB3    A0-B6       REGNO = IREG                       MCG       3352    A
MCG                                                                    MCG       3353    A
MCG      CRC     ROUTINE                                               MCG       3354    A
MCG              SA3    =22333333B                                     MCG       3355    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SA1    CRCA                                                    MCG        3356     A
MCG                SA2    RVT+RN.X0    I = RN.X0                                  MCG        3357     A
MCG                BX7    X3           RCOST = 22333333B                          MCG        3358     A
MCG                SB4    4            MCOST = 4                                  MCG        3359     A
MCG                SB5    B0           MAXD = 0                                   MCG        3360     A
MCG                ZR     B3,CRC4      IF REGNO = 0                               MCG        3361     A
MCG                                                                               MCG        3362     A
MCG                SA2    RVT+B3       I = REGNO                                  MCG        3363     A
MCG                SB6    B0           ABR = 0                                    MCG        3364     A
MCG                SX6    B3-RN.X0-6                                              MCG        3365     A
MCG                SX7    B1+B1                                                   MCG        3366     A
MCG                PL     X6,CRC4      RCOST = IF( REGNO \ X6 ) THEN 2 ELSE 3     MCG        3367     A
MCG                SX7    X7+B1                                                   MCG        3368     A
MCG                EQ     CRC4                                                    MCG        3369     A
MCG                                                                               MCG        3370     A
MCG       *        SEARCH *RVT* TO FIND RESULT WHICH IS CHEIPEST TO RESTORE,      MCG        3371     A
MCG       *        AND IS THE FURTHEST DISTANCE TO ITS NEXT USE.                  MCG        3372     A
MCG                                                                               MCG        3373     A
MCG        CRC1    MX4    -3                                                      MCG        3374     A
MCG                BX6    -X4*X7       COST = RCOST & 7                           MCG        3375     A
MCG                SB2    X6+                                                     MCG        3376     A
MCG                                                                               MCG        3377     A
MCG        CRC2    PX6    B2,X2        COST[RVT(I)] = COST                        MCG        3378     A
MCG                SA6    A2                                                      MCG        3379     A
MCG                GT     B2,B4,CRC3   IF COST > MCOST                            MCG        3380     A
MCG                LX2    -RV.NUP                                                 MCG        3381     A
MCG                SB3    X2                                                      MCG        3382     A
MCG                ZR     B3,CRC3      IF NU[RVT(I)] = 0  */ RESULT NOT USED YET  MCG        3383     A
MCG                LT     B2,B4,CRC2A  IF COST < MCOST                            MCG        3384     A
MCG                LT     B3,B5,CRC3   IF NU[RVT(I)] < MAXD                       MCG        3385     A
MCG        CRC2A   SB4    B2           MCOST = COST                               MCG        3386     A
MCG                SA0    A2           IREG = I                                   MCG        3387     A
MCG                SB5    B3           MAXD = NU[RVT(I)]                          MCG        3388     A
MCG                                                                               MCG        3389     A
MCG        CRC3    AX7    3            RCOST = SHIFT( RCOST , -3 )                MCG        3390     A
MCG                SA2    A2+B1        I = I + 1                                  MCG        3391     A
MCG                ZR     X7,CRC0      IF RCOST = 0                               MCG        3392     A
MCG                                                                               MCG        3393     A
MCG        CRC4    MI     X2,CRC3      IF LOCK[RVT(I)]                            MCG        3394     A
MCG                SB3    "TB"-B1                                                 MCG        3395     A
MCG                S"DI"  X2+B3        DI = BASE-1 + RI[RVT(I)]                   MCG        3396     A
MCG                ZR     X2,CRC3      IF RVT(I) = 0                              MCG        3397     A
MCG                BX6    X1*X4                                                   MCG        3398     A
MCG                LX4    59-D.RFP                                                MCG        3399     A
MCG                ZR     X6,CRC1      IF (LD[DI]&ZP[DI]&RF[DI]) = 0              MCG        3400     A
MCG                PL     X4,CRC5      IF ^RF[DI]                                 MCG        3401     A
MCG                IFNE   RF.SAV,0                                                MCG        3402     A
MCG                                                                               MCG        3403     A
MCG                ZR     B6,CRC5      IF ABR = 0    */ NO B-REGS AVAIL           MCG        3404     A
MCG                SB4    -B1          MCOST = -1                                 MCG        3405     A
MCG                SA0    A2           IREG = I                                   MCG        3406     A
MCG                EQ     CRC0                                                    MCG        3407     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                ENDIF                                                    MCG        3408     A
MCG                                                                         MCG        3409     A
MCG       CRC5     LX4    D.RFP-D.LDP                                       MCG        3410     A
MCG                SB2    B0          COST = 0                              MCG        3411     A
MCG                PL     X4,CRC2     IF ^LD[DI]    */ MUST BE ZEROP        MCG        3412     A
MCG                                                                         MCG        3413     A
MCG      *         INSTRUCTION IS A *LD*, SEE IF IT CAN BE REISSUED        MCG        3414     A
MCG      *         IF IT HAS OPERANDS, SEE IF THEY ARE IN B-REGISTERS.     MCG        3415     A
MCG                                                                         MCG        3416     A
MCG                SA3    CRCB                                              MCG        3417     A
MCG                BX6    X3*X4                                             MCG        3418     A
MCG                LX4    D.LDP-D.ZPP                                       MCG        3419     A
MCG                NZ     X6,CRC1     IF FP[DI] ! L2[DI]                    MCG        3420     A
MCG                MI     X4,CRC9     IF ZP[DI]     */ NO REG PRED          MCG        3421     A
MCG                LX4    1+D.ZPP+58-D.TYP                                  MCG        3422          I
          -FCC0584
MCG                PL     X4,CRC6     IF TYPE[DI] = I                       MCG        3423          I
          -FCC0584
MCG                BX6    X4                                                FCC0584       9     A
MCG                LX6    1+D.ZPP+58-D.TYP                                  FCC0584      10     A
MCG                PL     X6,CRC6     IF TYPE[DI] = I                       FCC0584      11     A
MCG                SA3    "DI"-B1     R2 = DI - 1                           MCG        3424     A
MCG                LX3    -IH.RFP                                           MCG        3425     A
MCG                EQ     CRC7                                              MCG        3426     A
MCG                                                                         MCG        3427     A
MCG       CRC6     SA3    "DI"-2      R1 = DI - 2                           MCG        3428     A
MCG                LX3    -R1.RJP                                           MCG        3429     A
MCG                                                                         MCG        3430     A
MCG       CRC7     BX6    "RN"X3      RN = RI[RWORD]                        MCG        3431     A
MCG                SA5    "TB"+X6     L = BASE + R                          MCG        3432     A
MCG                LX5    -I.REGP                                           MCG        3433     A
MCG                SX3    X5-8                                              MCG        3434     A
MCG                PL     X3,CRC1     IF REG[L] > 7 */ PRED NOT IN A B-REG  MCG        3435     A
MCG                SA5    LRN+X5                                            MCG        3436     A
MCG                BX6    X6-X5                                             MCG        3437     A
MCG                NZ     X6,CRC1     IF LRN( REG[L] ) " RN                 MCG        3438     A
MCG       CRC      SPACE  3,14                                              MCG        3439     A
MCG      *         CHECK ISSUED STORE CHAIN FOR AN INTERFERING STORE       MCG        3440     A
MCG                                                                         MCG        3441     A
MCG       CRC9     LX4    D.ZPP-D.EQVP                                      MCG        3442     A
MCG                SB2    B1          COST = 1                              MCG        3443     A
MCG                PL     X4,CRC2     IF ^EQV[DI]                           MCG        3444     A
MCG                SA3    ISC         J = ISC                               MCG        3445     A
MCG                SA5    "DI"-1      R2 = DI - 1                           MCG        3446     A
MCG                                                                         MCG        3447     A
MCG      *         IF NEXT USE IS AN EQUIV LINK TO A STORE, THEN CANNOT RELOAD MCG    3448     A
MCG                                                                         MCG        3449     A
MCG                LX2    -RV.NUP                                           MCG        3450     A
MCG                SA4    B3+X2       DS = BASE-1 + NU[RVT(I)]              MCG        3451     A
MCG                LX2    RV.NUP                                            MCG        3452     A
MCG                LX4    59-D.STP                                          MCG        3453     A
MCG                PL     X4,CRC10    IF ^ST[DS]    */ NEXT USE NOT IN A STORE MCG       3454     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SA4    A4-B1       R2S = DS - 1                                  MCG          3455     A
MCG                LX4    -IH.RFP                                                   MCG          3456     A
MCG                BX6    X2-X4                                                     MCG          3457     A
MCG                BX4    "RN"X6                                                    MCG          3458     A
MCG                NZ     X4,CRC1     IF RF[R2S] " RI[RVT(I)]  */ AN EQUIV LINK     MCG          3459     A
MCG                                                                                MCG          3460     A
MCG       CRC10    SX6    X3                                                        MCG          3461     A
MCG                ZR     X6,CRC1     IF J = 0                                      MCG          3462          I
          -FCC0584
MCG                ZR     X6,CRC2     IF J = 0      */ END OF CHAIN                 FCC0584        12     A
MCG                SA3    "TB"+X3     L  = BASE + J;  J = [L]                       MCG          3463     A
MCG                SA4    A3-2        R2J = L - 2                                   MCG          3464     A
MCG                BX6    X4-X5                                                     MCG          3465     A
MCG                LX6    -IH.IHP                                                   MCG          3466     A
MCG                SB3    X6                                                        MCG          3467     A
MCG                NZ     B3,CRC10    IF IH[R2] " IH[R2J]                           MCG          3468     A
MCG                LX6    IH.IHP-IH.RFP                                             MCG          3469     A
MCG                SB3    X6                                                        MCG          3470     A
MCG                NZ     B3,CRC1     IF RF[R2] " RF[R2J]                           MCG          3471          I
          -FCC1656X
MCG                NZ     B3,CRC11    IF RF[R2] " RF[R2J]                           FCC1656X        5     A
MCG                LX6    IH.RFP-IH.CAP                                             MCG          3472     A
MCG                SB3    X6                                                        MCG          3473     A
MCG                NZ     B3,CRC10    IF CA[R2] " CA[R2J]                           MCG          3474     A
MCG                                                                                FCC1656X        6     A
MCG       CRC11    SX4    A3-"TB"                                                   FCC1656X        7     A
MCG                SX6    X2+                                                       FCC1656X        8     A
MCG                IX6    X4-X6                                                     FCC1656X        9     A
MCG                MI     X6,CRC10    IF J < RI[RVT(I)]  */ ST BEFORE LD            FCC1656X       10     A
MCG                                                                                FCC1656X       11     A
MCG      *         ISSUE OF AN INTERFERING STORE COUNTS AS A USE, UPDATE *NU*      FCC1656X       12     A
MCG                                                                                FCC1656X       13     A
MCG                LX2    -RV.NUP                                                   FCC1656X       14     A
MCG                SX6    X2                                                        FCC1656X       15     A
MCG                IX5    X4-X6                                                     FCC1656X       16     A
MCG                LX2    RV.NUP                                                    FCC1656X       17     A
MCG                MI     X5,CRC1     IF NU[RVT(I)] > J                             FCC1656X       18     A
MCG                                                                                FCC1656X       19     A
MCG                SA5    "TB"+X2     LI = BASE + RI[RVT(I)]                        FCC1656X       20     A
MCG                SA4    O.TREE                                                    FCC1656X       21     A
MCG                MX3    -I.INDXL                                                  FCC1656X       22     A
MCG                LX5    -I.INDXP                                                  FCC1656X       23     A
MCG                SB3    X4-1                                                      FCC1656X       24     A
MCG                BX3    -X3*X5                                                    FCC1656X       25     A
MCG                SA4    B3+X3       UI = TREE-1 + INDX[LI]                        FCC1656X       26     A
MCG                                                                                FCC1656X       27     A
MCG       CRC12    SA4    A4+B1       UI = UI + 1                                   FCC1656X       28     A
MCG                LX4    -U.SUCCP                                                  FCC1656X       29     A
MCG                SA5    "TB"+X4     SI = BASE + SUCC[UI]                          FCC1656X       30     A
MCG                LX5    59-I.IIP                                                  FCC1656X       31     A
MCG                MI     X5,CRC12    IF II[SI]                                     FCC1656X       32     A
MCG                                                                                FCC1656X       33     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                LX2    -RV.NUP                                          FCC1656X      34     A
MCG                SX4    X4                                               FCC1656X      35     A
MCG                BX2    X2-X6                                            FCC1656X      36     A
MCG                BX6    X2+X4       NU[RVT(I)] = SUCC[UI]                FCC1656X      37     A
MCG                LX6    RV.NUP                                           FCC1656X      38     A
MCG                BX2    X6                                               FCC1656X      39     A
MCG                SA6    A2                                               FCC1656X      40     A
MCG                EQ     CRC1                                             MCG         3475     A
MCG                                                                        MCG         3476     A
MCG                IFEQ   RF.SAV,0,2                                       MCG         3477     A
MCG       CRCA     BFMW   D,(LD,ZP)                                        MCG         3478     A
MCG                ELSE   1                                                MCG         3479     A
MCG       CRCA     BFMW   D,(LD,ZP,RF)                                     MCG         3480     A
MCG       CRCB     BFMW   D,(L2,FP),59-D.LDP                               MCG         3481     A
MCG       CRA      TITLE  CRA - CHANGE REGISTER ASSIGNMENT                 MCG         3482     A
MCG      **        CRA - CHANGE REGISTER ASSIGNMENT                        MCG         3483     A
MCG      *                                                                 MCG         3484     A
MCG      *         THOSE WISHING TO EXPRESS THEIR APPREATION FOR THE EXISTANCE OF  MCG  3485     A
MCG      *         THIS ROUTINE AND OTHERS WHICH GENERATE *XMT"S* , ETC MAY  MCG  3486     A
MCG      *         THANK SEYMOUR CRAY.                                     MCG         3487     A
MCG      *                                                                 MCG         3488     A
MCG      *         ENTRY  AVAIL " 0 , AVAIL & CLASS = 0                    MCG         3489     A
MCG      *                (X1) = CLASS                                     MCG         3490     A
MCG      *                (T0) = TXT INDEX OF INSTRUCTION THAT CANNOT BE ISSUED  MCG  3491     A
MCG                                                                        MCG         3492     A
MCG       CRA      ROUTINE                                                 MCG         3493     A
MCG                SX7    X1-FX.ST                                         MCG         3494     A
MCG                NZ     X7,CRA4     IF CLASS " FX.ST                     MCG         3495     A
MCG                                                                        MCG         3496     A
MCG      *         STORE REGISTERS JAMMED, FREE ONE OF THEM UP             MCG         3497     A
MCG                                                                        MCG         3498     A
MCG                SA1    AXR                                              MCG         3499     A
MCG                SX5    FX.SCR+FX.XT                                     MCG         3500     A
MCG                BX7    X1*X5       AVAIL = AXR & FX.SCR                 MCG         3501     A
MCG                NX6    B3,X7                                            MCG         3502     A
MCG                SA3    B3+FXRA+16-47                                    MCG         3503     A
MCG                SB6    "TB"-B1                                          MCG         3504     A
MCG                SB3    X3          REGR = FXRA( NORMC(AVAIL) )          MCG         3505     A
MCG                SB4    RN.X0+6                                          MCG         3506          I
          -FCC1056
MCG                SB4    RN.X0+6     REGF = X6                            FCC1056       34     A
MCG                SA1    RVT+B4      RX6 = RI[RVT(RN.X6)]                 MCG         3507     A
MCG                SA2    A1+B1       RX7 = RI[RVT(RN.X7)]                 MCG         3508     A
MCG                SA3    B6+X1                                            MCG         3509          I
          -FCC1056
MCG                SA4    B6+X2                                            MCG         3510          I
          -FCC1056
MCG                BX5    X3*X4                                            MCG         3511          I
          -FCC1056
MCG                LX5    59-D.PSP                                         MCG         3512          I
          -FCC1056
MCG                MI     X5,CRA2     IF PS[RX6] & PS[RX7]                 MCG         3513          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

          -FCC1056
MCG                 LX3    59-D.PSP                                                MCG        3514          I
          -FCC1056
MCG                 PL     X3,CRA1     REGF = IF( ^PS[RX6] ) THEN X6 ELSE X7       MCG        3515          I
          -FCC1056
MCG                 SB4    B4+B1                                                   MCG        3516          I
          -FCC1056
MCG                 SA3    B6+X1       DX6 = BASE-1 + RX6                          FCC1056      35     A
MCG                 SA4    B6+X2       DX7 = BASE-1 + RX7                          FCC1056      36     A
MCG                 MX7    -D.STRSL                                                FCC1056      37     A
MCG                 LX7    D.STRSP                                                 FCC1056      38     A
MCG                 BX5    -X7*X3                                                  FCC1056      39     A
MCG                 BX6    -X7*X4                                                  FCC1056      40     A
MCG                 ZR     X5,CRA1     IF STRS[DX6] = 0                            FCC1056      41     A
MCG                 NZ     X6,CRA2     IF STRS[DX7] " 0                            FCC1056      42     A
MCG                 SB4    B4+1        REGF = X7                                   FCC1056      43     A
MCG                                                                               MCG        3517     A
MCG       CRA1      SB2    OC.XMT                                                  MCG        3518     A
MCG                 RJ     OXI         OUTPUT XMT REGF TO REGR                     MCG        3519     A
MCG                 SA2    T0                                                      FCC0584      13     A
MCG                 RJ     ISI         ISSUE T0                                    FCC0584      14     A
MCG                 EQ     CRA                                                     MCG        3520     A
MCG                                                                               MCG        3521     A
MCG       *         BOTH RESULT IN X6 AND X7 PS, THIS IS A PATHOLOGICAL CASE.     MCG        3522          I
          -FCC1056
MCG       *         BOTH RESULTS IN X6 & X7 PS!PRS, THIS IS A PATHALOGICAL CASE.  FCC1056      44     A
MCG       *         U = A; V = B; IF(W) GO TO LAB; X(I) = C; X(J) = A; X(K) = B;  MCG        3523     A
MCG                                                                               MCG        3524     A
MCG       CRA2      LX1    -RV.NUP                                                 MCG        3525     A
MCG                 LX2    -RV.NUP                                                 MCG        3526     A
MCG                 SB2    X1          DX6 = NU[RVT(RN.X6)]                        MCG        3527     A
MCG                 SB5    X2          DX7 = NU[RVT(RN.X67]                        MCG        3528          I
          -FCC0393
MCG                 SB5    X2          DX7 = NU[RVT(RN.X7)]                        FCC0393      15     A
MCG                 GE     B2,B5,CRA3  REGF = IF( DX6 \ DX7 ) THEN RN.X6 ELSE X7   MCG        3529     A
MCG                 SB4    B4+B1                                                   MCG        3530     A
MCG                 BX1    X2                                                      MCG        3531     A
MCG                                                                               MCG        3532     A
MCG       CRA3      LX1    RV.NUP                                                  MCG        3533     A
MCG                 SB2    OC.XMT                                                  MCG        3534     A
MCG                 BX6    "RN"X1      R0 = RI[RVT(REGF)]                          MCG        3535     A
MCG                 PX7    B3,X6                                                   MCG        3536     A
MCG                 SA7    R0                                                      MCG        3537     A
MCG                 RJ     OXI         OUTPUT XMT REGF TO REGR                     MCG        3538     A
MCG                 SA2    T0                                                      MCG        3539     A
MCG                 RJ     ISI         ISSUE *T0*                                  MCG        3540     A
MCG                 RJ     SXB         SETUP AN XMT BACK TO THE STORE REG          MCG        3541          I
          -FCC0881
MCG                 SX6    B0                                                      FCC0881      38     A
MCG                 RJ     SXB         SETUP *XMT* BACK                            FCC0881      39     A
MCG                 EQ     CRA                                                     MCG        3542     A
MCG                                                                               MCG        3543     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG       *         *LD* REGISTER NEEDED.                                          MCG        3544     A
MCG                                                                                MCG        3545     A
MCG        CRA4     SX7    X1-FX.LD                                                MCG        3546     A
MCG                 NZ     X7,CRA10    IF CLASS " FX.LD                            MCG        3547     A
MCG                 SB3    B0                                                      MCG        3548     A
MCG                 SB6    B0          ABR = 0                                     MCG        3549     A
MCG                 RJ     CRC         COMPUTE RESTORE COSTS                       MCG        3550     A
MCG                 SA1    AXR                                                     MCG        3551     A
MCG                 SX2    FX.ST                                                   MCG        3552     A
MCG                 BX3    X2*X1                                                   MCG        3553     A
MCG                 ZR     X3,CRA9     IF AXR & FX.ST = 0  */ NO STORE REG AVAIL   MCG        3554     A
MCG                                                                                MCG        3555     A
MCG       *         STORE REG AVAIL, MOVE RESULT THAT HAS HIGHEST RESTORE COST     MCG        3556     A
MCG       *         AND IS FURTHEST DISTANCE TO A NEXT USE TO IT, ON THE ASSUMPTION MCG       3557     A
MCG       *         THAT IT MAY BE SAVED LATER.                                    MCG        3558     A
MCG                                                                                MCG        3559     A
MCG                 SB4    -B1         MAXC = -1                                   MCG        3560     A
MCG                 SB5    -B1         MAXD = -1                                   MCG        3561     A
MCG                 SB2    RN.X0+1     I = X1                                      MCG        3562     A
MCG                 SA0    B2          REGF = X1     */ INCASE ALL LD-REGS LOCKED  FCC0448     150     A
MCG                                                                                MCG        3563     A
MCG        CRA5     SA1    RVT+B2                                                  MCG        3564     A
MCG                 MI     X1,CRA7     IF LOCK[RVT(I)]                             MCG        3565     A
MCG                 IF     DEF,JXR,1                 */ TEST MODE ONLY             MCG        3566          I
          -FCC0625
MCG                 ZR     X1,CRA7     IF RVT(I) = 0                               MCG        3567          I
          -FCC0625
MCG                 SB6    "TB"-B1                                                 FCC0625     10     A
MCG                 SA4    B6+X1       DI = TB-1 + RI[RVT(I)]                      FCC0625     11     A
MCG                 LX4    59-D.PRSP                                               FCC0625     12     A
MCG                 MI     X4,CRA7     IF PRS[DI]                                  FCC0625     13     A
MCG                 UX6    B3,X1                                                   MCG        3568     A
MCG                 LX1    -RV.NUP                                                 MCG        3569     A
MCG                 GT     B3,B4,CRA6  IF COST[RVT(I)] > MAXC                      MCG        3570     A
MCG                 LT     B3,B4,CRA7  IF COST[RVT(I)] < MAXC                      MCG        3571     A
MCG                 SB6    X1+                                                     MCG        3572     A
MCG                 LT     B6,B5,CRA7  IF NU[RVT(I)] < MAXD                        MCG        3573     A
MCG        CRA6     SA0    B2          REGF = I                                    MCG        3574     A
MCG                 SB4    B3          MAXC = COST[RVT(I)]                         MCG        3575     A
MCG                 SB5    B6          MAXD = NU[RVT(I)]                           MCG        3576     A
MCG        CRA7     SB2    B2+1        I = I + 1                                   MCG        3577     A
MCG                 SX6    B2-RN.X0-6                                              MCG        3578     A
MCG                 MI     X6,CRA5     IF I < RN.X6                                MCG        3579     A
MCG                 SB3    RN.X0+6                                                 MCG        3580     A
MCG                 SA1    RVT+B3                                                  MCG        3581     A
MCG                 SB4    A0                                                      MCG        3582     A
MCG                 SB6    "TB"-B1                                                 MCG        3583     A
MCG                 ZR     X1,CRA8     REGR = IF( RVT(X6) = 0 ) THEN X6 ELSE X7    MCG        3584     A
MCG                 SB3    B3+B1                                                   MCG        3585     A
MCG                                                                                MCG        3586     A
MCG        CRA8     SA1    RVT+B4                                                  MCG        3587     A
MCG                 S"DI"  B6+X1       DI = BASE-1 + RI[RVT(REGF)]                 MCG        3588     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                LX4    59-D.PRSP                                          MCG        3589    A
MCG                PL     X4,CRA1     IF ^PRS[DI]                            MCG        3590    A
MCG                LX1    -RV.NUP                                            MCG        3591    A
MCG                SA2    RBV+B4                                             FCC0881      40    A
MCG                SA3    LXR                                                FCC0881      41    A
MCG                BX6    -X2*X3      LXR = LXR & ^RBV(REGF)  */ CLEAR LOCK BIT FCC0881   42    A
MCG                SA6    A3                                                 FCC0881      43    A
MCG                EQ     CRA3                                               MCG        3592    A
MCG         CRA    SPACE  2,10                                               MCG        3593    A
MCG        *       LD REG NEEDED AND NO STORE REG AVAILABLE                  MCG        3594          I
           -FCC0502
MCG                                                                          MCG        3595          I
           -FCC0502
MCG         CRA9   SB4    B3                                                 MCG        3596          I
           -FCC0502
MCG                SB3    RN.X0       REGR = RN.X0                           MCG        3597          I
           -FCC0502
MCG                SB6    "TB"-B1                                            MCG        3598          I
           -FCC0502
MCG                EQ     CRA8                                               MCG        3599          I
           -FCC0502
MCG        *       LD REG NEEDED AND NO ST REG AVAILABLE, SEARCH FOR ^PRS LD-REG. FCC0502     5    A
MCG                                                                          FCC0502      6    A
MCG         CRA9   SB3    RN.X0       REGR = RN.X0  */ RESULT REG           FCC0502      7    A
MCG                SB6    "TB"-B1                                            FCC0502      8    A
MCG                SB4    B3+B1       REGF = RN.X1  */ OPERAND               FCC0502      9    A
MCG                                                                          FCC0502     10    A
MCG         CRA9A  SA1    RVT+B4                                             FCC0502     11    A
MCG                S"DI"  B6+X1       DI = TB-1 + RN[RVT(REGF)]              FCC0502     12    A
MCG                LX4    59-D.PRSP                                          FCC0502     13    A
MCG                PL     X4,CRA1     IF ^PRS[DI]                            FCC0502     14    A
MCG                SB4    B4+B1       REGF = REGF + 1                        FCC0502     15    A
MCG                SX6    B4-RN.X0-5                                         FCC0502     16    A
MCG                MI     X6,CRA9A    IF REGF < RN.X5                        FCC0502     17    A
MCG                                                                          FCC0502     18    A
MCG                EQ     CRA8                                               FCC0502     19    A
MCG                                                                          MCG        3600    A
MCG        *       INSTRUCTION PRS AND CANNOT BE ISSUED                      MCG        3601    A
MCG                                                                          MCG        3602    A
MCG         CRA10  SA1    T0                                                 MCG        3603    A
MCG                SB6    "TB"+B1                                            MCG        3604    A
MCG                SA2    X1+B6       R1RS = BASE + T0+4                     MCG        3605    A
MCG                MX5    -SO.REGL                                           MCG        3606    A
MCG                LX2    -R1.SOP                                            MCG        3607    A
MCG                BX6    -X5*X2                                             MCG        3608    A
MCG                SB4    X6          REGF = SOREG[R1RS]                     MCG        3609    A
MCG                SX7    X6-8                                               MCG        3610    A
MCG                MI     X7,CRA11    IF REGF @ 7   */ A B-REGISTER          MCG        3611    A
MCG                SA1    AXR                                                MCG        3612    A
MCG                SX5    FX.SCR+FX.XT                                       MCG        3613    A
MCG                BX7    X1*X5       AVAIL = AXR & FX.SCR                   MCG        3614    A
MCG                NX6    B3,X7                                              MCG        3615    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SA3    B3+FXRA+16-47                                    MCG          3616     A
MCG                SB3    X3          REGR = FXRA( NORMC(AVAIL) )          MCG          3617     A
MCG                EQ     CRA1                                             MCG          3618          I
          -MCGF1
MCG                SA1    RVT+B4                                           MCGF1           5     A
MCG                SB6    "TB"-B1                                          MCGF1           6     A
MCG                SA2    B6+X1       DI = BASE-1 + RN[RVT(REGF)]          MCGF1           7     A
MCG                LX2    59-D.PSP                                         MCGF1           8     A
MCG                PL     X2,CRA1     IF ^PS[DI]                           MCGF1           9     A
MCG                LX1    -RV.NUP                                          MCGF1          10     A
MCG                EQ     CRA3                                             MCGF1          11     A
MCG                                                                       MCG          3619     A
MCG       CRA11    EQU    UJR.E1      B-REGISTER DEADLOCK                  MCG          3620          I
          -FCC1555X
MCG       *        INSTRUCTION PRS TO A B-REGISTER AND IT CANNOT BE ISSUED, SEE   FCC1555X    20     A
MCG       *        IF THE USES OF THE R-NUMBER IN THE B-REGISTER ARE SUCH THAT    FCC1555X    21     A
MCG       *        WE CAN MOVE IT TO AN X-REGISTER.                               FCC1555X    22     A
MCG                                                                       FCC1555X    23     A
MCG       CRA11    SA1    RVT+B4                                           FCC1555X    24     A
MCG                SA2    JLR                                              FCC1555X    25     A
MCG                SB3    B4+12                                            FCC1555X    26     A
MCG                LX2    B3                                               FCC1555X    27     A
MCG                BX3    X1+X2                                            FCC1555X    28          I
          -FCC1845X
MCG                MI     X3,CRA15    IF LOCK[RVT(REGF)] ! SHIFT(JLR,REGF+12) < 0    FCC1555X    29          I
          -FCC1845X
MCG                MI     X2,CRA15    IF SHIFT(JLR,REG+12) < 0 */ JUP LOCK           FCC1845X     7     A
MCG                SA2    "TB"+X1     LI = TB + RN[RVT(REGF)]              FCC1555X    30     A
MCG                MX7    -I.INDXL                                         FCC1555X    31     A
MCG                SA3    O.TREE                                           FCC1555X    32     A
MCG                LX2    -I.INDXP                                         FCC1555X    33     A
MCG                BX6    -X7*X2                                           FCC1555X    34     A
MCG                SB3    X3                                               FCC1555X    35     A
MCG                SA3    B3+X6       UI = TREE + INDX[LI]                 FCC1555X    36     A
MCG                EQ     CRA13                                            FCC1555X    37     A
MCG                                                                       FCC1555X    38     A
MCG       CRA12    LX3    59-U.LASTP+U.SUCCP                               FCC1555X    39     A
MCG                MI     X3,CRA21    IF LAST[UI]                          FCC1555X    40     A
MCG                SA3    A3+1        UI = UI + 1                          FCC1555X    41     A
MCG                                                                       FCC1555X    42     A
MCG       CRA13    LX3    -U.SUCCP                                         FCC1555X    43     A
MCG                SA5    "TB"+X3     LIS = TB + SUCC[UI]                  FCC1555X    44     A
MCG                LX5    59-I.IIP                                         FCC1555X    45     A
MCG                MI     X5,CRA12    IF II[LIS]    */ ISSUED              FCC1555X    46     A
MCG                SA5    A5-B1       DIS = LIS - 1                        FCC1555X    47     A
MCG                SA2    A5-2                                             FCC1655X     4     A
MCG                LX6    B1,X5                                            FCC1555X    48     A
MCG                ERRNZ  59-D.TYP-1                                       FCC1555X    49     A
MCG                MI     X6,CRA12    IF TYPE[DIS] = II ! IV               FCC1555X    50     A
MCG                LX5    58-D.TYP                                         FCC1555X    51     A
MCG                PL     X5,CRA14    IF TYPE[DIS] = I                     FCC1555X    52     A
MCG                LX5    1+D.TYP-D.JPP                                    FCC1555X    53     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                PL     X5,CRA12    IF ^JP[DIS]   */ ^ B USE                    FCC1555X    54    A
MCG                EQ     CRA15                                                   FCC1555X    55    A
MCG                                                                               FCC1555X    56    A
MCG      *         TYPE I INST, B-RESULT IS USED IN RJ OF ILS, IRS AND PK, AND     FCC1555X    57    A
MCG      *         RJ, RK OF SA, SS, SLD, SST, SDL, SDS. SUFFICIENT TO CHECK       FCC1555X    58    A
MCG      *         FOR RK[R1] = 0 TO IMPLY X-USE POSSIBLE                          FCC1555X    59    A
MCG                                                                               FCC1555X    60    A
MCG       CRA14    SA5    A5-2        R1S = DIS - 2                               FCC1555X    61    A
MCG                UX7    B3,X5                                                   FCC1655X     5    A
MCG                LX5    -R1.RKP                                                 FCC1555X    62    A
MCG                BX6    "RN"X5                                                  FCC1555X    63    A
MCG                ZR     X6,CRA12    IF RK[R1S] = 0  */ CAN MOVE TO AN X-REGISTER FCC1555X    64    A
MCG                SB5    OC.SLD                                                  FCC1655X     6    A
MCG                LT     B3,B5,CRA15 IF OC[R1] < OC.SLD                          FCC1655X     7    A
MCG                SB5    OC.SA                                                   FCC1655X     8    A
MCG                GT     B3,B5,CRA15 IF OC[R1] > OC.SA  */ NOT SLD, SST OR SA    FCC1655X     9    A
MCG                                                                               FCC1655X    10    A
MCG      *         IF BOTH OPERANDS IN A B-REGISTER, THEN ONE CAN BE MOVED         FCC1655X    11    A
MCG                                                                               FCC1655X    12    A
MCG                SX1    X1          R = RN[RVT(REGF)]  */ RI IN B-REG           FCC1655X    13    A
MCG                BX7    X1-X6       ORN = RK[R1]                                FCC1655X    14    A
MCG                NZ     X7,CRA14A   IF R " ORN                                  FCC1655X    15    A
MCG                LX5    R1.RKP-R1.RJP                                           FCC1655X    16    A
MCG                BX6    "RN"X5      ORN = RJ[R1]                                FCC1655X    17    A
MCG       CRA14A   SB5    "TB"-3                                                  FCC1655X    18    A
MCG                SA5    B5+X6       R1P = TXT(ORN)                              FCC1655X    19    A
MCG                UX7    B3,X5                                                   FCC1655X    20    A
MCG                SB5    OC.DEF                                                  FCC1655X    21    A
MCG                NE     B3,B5,CRA15 IF OC[R1P] " OC.DEF                         FCC1655X    22    A
MCG                LX5    -R1.SOP-SO.RTP                                          FCC1655X    23    A
MCG                MX7    -SO.RTL                                                 FCC1655X    24    A
MCG                BX6    -X7*X5                                                  FCC1655X    25    A
MCG                ZR     X6,CRA21    IF RT[R1P] = 0  */ A B-REGISTER             FCC1655X    26    A
MCG                                                                               FCC1555X    65    A
MCG      *         INSTRUCTION HAS B USES THAT CANNOT BE CHANGED TO X USES.        FCC1555X    66    A
MCG                                                                               FCC1555X    67    A
MCG       CRA15    EQ     UJR.E1                                                  FCC1555X    68    A
MCG                                                                               FCC1555X    69    A
MCG      *         ALL FUTURE USES OF INSTRUCTION CAN BE X-USES, MOVE INSTRUCTION  FCC1555X    70    A
MCG      *         TO AN X-REGISTER.                                               FCC1555X    71    A
MCG                                                                               FCC1555X    72    A
MCG       CRA21    SA1    AXR                                                     FCC1555X    73    A
MCG                SA2    T                                                       FCC1555X    74    A
MCG                SX5    FX.SCR+FX.XT                                            FCC1555X    75    A
MCG                BX7    X1*X5       AVAIL = AXR & FX.SCR                        FCC1555X    76    A
MCG                NX6    B3,X7                                                   FCC1555X    77    A
MCG                SB5    X2                                                      FCC1555X    78    A
MCG                SA3    B3+FXRA+16-47                                           FCC1555X    79    A
MCG                SA2    TRA+X3                                                  FCC1555X    80    A
MCG                SB2    X2                                                      FCC1555X    81    A
MCG                SB3    X3          REGR = FXR( SCR )                           FCC1555X    82    A
MCG                LE     B2,B5,CRA22 T(IS) = MAX( T , T(IS) )                    FCC1555X    83    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SB5    B2                                                    FCC1555X      84     A
MCG       CRA22    SB2    OC.SA                                                 FCC1555X      85     A
MCG                RJ     OXI         OUTPUT A *SA*                             FCC1555X      86     A
MCG                SA2    T0                                                    FCC1555X      87     A
MCG                                                                             CCG0191        7     A
MCG      *         CHECK IF REGR OF *SA* JUST ISSUED IS IN K FIELD OF T0.        CCG0191        8     A
MCG      *         IF IT IS, SWITCH J AND K.                                     CCG0191        9     A
MCG                                                                             CCG0191       10     A
MCG                SX2    X2-3                            */TXT INDEX OF T0      CCG0191       11     A
MCG                SA5    "TB"+X2                                               CCG0191       12     A
MCG                UX7    B2,X5                                                 CCG0191       13     A
MCG                SB2    B2-OC.SA                                              CCG0191       14     A
MCG                NZ     B2,CRA24             */ IF OC OF T0 NOT *SA*          CCG0191       15     A
MCG                LX7    -R1.RKP                                               CCG0191       16     A
MCG                BX6    "RN"X7                                                CCG0191       17     A
MCG                BX3    -X6                                                   CCG0191       18     A
MCG                SX3    X3+B3                                                 CCG0191       19     A
MCG                NZ     X3,CRA24       */IF REGR OF *SA* NOT IN K FIELD OF T0 CCG0191       20     A
MCG                LX6    R1.RJP                                */ K TO J       CCG0191       21     A
MCG                LX7    R1.RKP-R1.RJP                                         CCG0191       22     A
MCG                BX7    "RN"X7                                                CCG0191       23     A
MCG                LX7    R1.RKP                                 */ J TO K      CCG0191       24     A
MCG                MX1    R1.OCL+R1.RIL                                         CCG0191       25     A
MCG                BX6    X6+X7                                     */ J + K    CCG0191       26     A
MCG                LX1    R1.RIL                                                CCG0191       27     A
MCG                BX5    X1*X5                           */CLEAR OLD *JK*      CCG0191       28     A
MCG                BX6    X6+X5                           */ADD SWITCHED *JK*   CCG0191       29     A
MCG                SA6    A5                                                    CCG0191       30     A
MCG                                                                             CCG0191       31     A
MCG       CRA24    SA2    T0                                                    CCG0191       32     A
MCG                RJ     ISI         ISSUE T0                                  FCC1555X      88     A
MCG                EQ     CRA                                                   FCC1555X      89     A
MCG       SXB      SPACE  3           SXB                                       MCG         3621          I
         -FCC1056
MCG       SXB      EJECT                                                        FCC1056       45     A
MCG      **        SXB - SETUP XMT BACK ( INSTRUCTION PS/PRS )                  MCG         3622     A
MCG      *                                                                      MCG         3623     A
MCG      *         ENTRY  (R0) = *TXT* INDEX OF INSTRUCTION                     MCG         3624     A
MCG      *                (X6) = CALLF = 0 IF CALL FROM *CRA*                   FCC0881       44     A
MCG                                                                             MCG         3625     A
MCG       SXB      ROUTINE                                                      MCG         3626     A
MCG                SA2    R0                                                    FCC0881       45     A
MCG                SB6    "TB"-B1                                               FCC0881       46     A
MCG                SA4    B6+X2          DI = BASE-1 + R0                       FCC0881       47     A
MCG                LX4    59-D.PSP                                              FCC0881       48     A
MCG                MI     X4,SXBS        IF PS[DI]                              FCC0881       49     A
MCG                                                                             FCC0881       50     A
MCG                SA5    L.TXT          LT = L.TXT                             MCG         3627     A
MCG                ALLOC  TXT,4                                                 MCG         3628     A
MCG                SA4    R0                                                    MCG         3629     A
MCG                UX6    B3,X4                                                 MCG         3630     A
MCG                BX7    X5                                                    MCG         3631     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SA7    RVT+B3      RVT(REGR) = LT                          MCG          3632     A
MCG                BX4    "RN"X4                                              MCG          3633     A
MCG                IX3    X2+X5                                               MCG          3634     A
MCG                IX2    X2+X4                                               MCG          3635     A
MCG                MOVE   X1,X2,X3    MOVE( 4 , O.TXT+R0 , O.TXT+LT )         MCG          3636     A
MCG                RJ     AIP         ADJUST POINTERS                         MCG          3637     A
MCG                                                                           MCG          3638     A
MCG       *        SETUP AN *XMT* WHERE THE ORIGINAL INSTRUCTION WAS          MCG          3639     A
MCG                                                                           MCG          3640     A
MCG                SA1    R0                                                  MCG          3641     A
MCG                SB2    OC.XMT                                              MCG          3642     A
MCG                SB6    "TB"-B1                                             MCG          3643     A
MCG                SA2    F.RDT+B2                                            MCG          3644     A
MCG                MX6    -D.STRSL                                            MCG          3645     A
MCG                S"DI"  B6+X1       D0 = BASE-1 + R0                        MCG          3646     A
MCG                LX6    D.STRSP                                             MCG          3647     A
MCG                MX7    0                                                   MCG          3648     A
MCG                SA7    "DI"-B1     [DI-1] = 0                              MCG          3649     A
MCG                BX4    -X6*X4                                              MCG          3650     A
MCG                LX5    R1.RJP                                              MCG          3651     A
MCG                BX5    X5+X1                                               MCG          3652     A
MCG                PX6    B2,X5                                               MCG          3653     A
MCG                SA6    A7-B1       [DI-2] = TYI(OC.XMT,LT,0,R0)            MCG          3654     A
MCG                BX7    X4+X2                                               MCG          3655     A
MCG                SA7    A4          [DI] = RDT(OC.XMT) ! STRS[DI]           MCG          3656     A
MCG                SA5    A4+B1                                               FCC0881        51     A
MCG                MX3    1                                                   FCC0881        52     A
MCG                LX3    1+I.IIP                                             FCC0881        53     A
MCG                BX6    -X3*X5      II[DI+1] = 0  */ CLEAR IN CASE STORED OUT FCC0881        54     A
MCG                SA6    A5                                                  FCC0881        55     A
MCG                RJ     RIL         RESET THE ISSUE LIST                    MCG          3657     A
MCG                SA3    SIW                                                 MCG          3658     A
MCG                NZ     X3,SXB1     IF SIW " 0                              MCG          3659          I
          -FCC0881
MCG                NZ     X3,SXB0     IF SIW " 0                              FCC0881        56     A
MCG                ALLOC  TREE,1                                              MCG          3660     A
MCG                SX3    X3-1        SIW = L.TREE - 1                        MCG          3661     A
MCG                RJ     AIP         ADJUST *PIT* POINTERS                   MCG          3662     A
MCG                                                                           MCG          3663     A
MCG        SXB1    SA1    L.TXT                                               MCG          3664          I
          -FCC0881
MCG        SXB0    SA1    L.TXT                                               FCC0881        57     A
MCG                SA2    SXBA                                                MCG          3665     A
MCG                SA4    O.TREE                                              MCG          3666     A
MCG                SB4    X1-4                                                MCG          3667     A
MCG                SA5    "TB"+B4     L = BASE + L.TXT-4                      MCG          3668     A
MCG                SB2    X3                                                  MCG          3669     A
MCG                BX6    -X2*X5                                              MCG          3670     A
MCG                LX3    I.INDXP                                             MCG          3671     A
MCG                SX7    B1                                                  MCG          3672     A
MCG                BX6    X6+X3       INDX[L] = SIW                           MCG          3673     A
MCG                LX7    I.USESP                                             MCG          3674     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                IX6    X6+X7       USES[L] = 1                                  MCG          3675     A
MCG                SA2    R0                                                       FCC0881        58     A
MCG                MX5    -I.LSTL                                                  FCC0881        59     A
MCG                SA3    "TB"+X2     LI = BASE + R0                               FCC0881        60     A
MCG                LX5    I.LSTP                                                   FCC0881        61     A
MCG                BX3    -X5*X3      LST = LST[LI] */ LST OF *XMT* BACK           FCC0881        62     A
MCG                BX6    X3+X6       LST[L] = LST                                 FCC0881        63     A
MCG                SA6    A5                                                       MCG          3676     A
MCG                MX1    -D.STRSL                                                 FCC0625        14     A
MCG                SA5    A5-B1       DI = L - 1                                   FCC0625        15     A
MCG                LX1    D.STRSP                                                  FCC0625        16     A
MCG                BX6    X1*X5       STRS[DI] = 0  */ CLEAR *STRS* OF MOVED INST  FCC0625        17     A
MCG                SA6    A5                                                       FCC0625        18     A
MCG                                                                               MCG          3677     A
MCG                LX7    U.LASTP-I.USESP                                          MCG          3678     A
MCG                SA1    R0                                                       MCG          3679          I
          -FCC0881
MCG                BX7    X7+X1                                                    MCG          3680          I
          -FCC0881
MCG                SA7    X4+B2       [O.TREE+SIW] = SIT(1,0,0,R0)                 MCG          3681          I
          -FCC0881
MCG                LX3    U.LSTP-I.LSTP                                            FCC0881        64     A
MCG                SX2    X2                                                       FCC0881        65     A
MCG                BX3    X3+X2                                                    FCC0881        66     A
MCG                BX7    X7+X3                                                    FCC0881        67     A
MCG                SA7    X4+B2       TREE(SIW) = SIT(1,0,LST,R0)                  FCC0881        68     A
MCG                EQ     SXB                                                      MCG          3682     A
MCG                                                                               FCC0881        69     A
MCG      *         INSTRUCTION *PS*, NOT ALL USES MAY BE STORES, SETUP *XMT LT,R0  FCC0881        70     A
MCG      *         AT END OF *TXT*, CHANGE *RI"S* IN THE ST"S, ^ST USES WILL WORK  FCC0881        71     A
MCG      *         OFF ORIGINAL *RI* ( R0 ).                                       FCC0881        72     A
MCG                                                                               FCC0881        73     A
MCG       SXBS     SA6    SXBB                                                     FCC0881        74     A
MCG                SA2    R0                                                       FCC0881        75     A
MCG                SA1    L.TXT                                                    FCC0881        76     A
MCG                SA4    O.TREE                                                   FCC0881        77     A
MCG                SA5    "TB"+X2     LI = BASE + R0                               FCC0881        78     A
MCG                MX7    -I.INDXL                                                 FCC0881        79     A
MCG                SB3    B0          UST = 0       */ N.ST USES                   FCC0881        80     A
MCG                SB4    X4                                                       FCC0881        81     A
MCG                SB5    B0          FSU = 0       */ FIRST ST USE                FCC0881        82     A
MCG                LX5    -I.INDXP                                                 FCC0881        83     A
MCG                BX6    -X7*X5                                                   FCC0881        84     A
MCG                SA3    B4+X6       UI = TREE + INDX[LI]                         FCC0881        85     A
MCG                BX5    "RN"X2                    (X5) = R0                      FCC1126         7     A
MCG                EQ     SXB2                                                     FCC0881        86     A
MCG                                                                               FCC0881        87     A
MCG      *         SCAN *SIT*, COUNT USES OF *R0* AS A ST-PRED, CHANGE THEM TO     FCC0881        88     A
MCG                USE AN *XMT* AT THE END OF *TXT* AS THEIR *RI*.                 FCC0881        89          I
          -RSM2763X
MCG      *         USE AN *XMT* AT THE END OF *TXT* AS THEIR *RI*.                 RSM2763X      369     A
MCG                                                                               FCC0881        90     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG       SXB1     LX3    59-U.LASTP                                    FCC0881      91     A
MCG                MI     X3,SXB3     IF LAST[UI]                      FCC0881      92     A
MCG                SA3    A3+1        UI = UI + 1                      FCC0881      93     A
MCG                                                                    FCC0881      94     A
MCG       SXB2     LX3    -U.SUCCP                                     FCC0881      95     A
MCG                SA2    "TB"+X3     LIS = BASE + SUCC[UI]            FCC0881      96     A
MCG                LX3    U.SUCCP                                      FCC0881      97     A
MCG                LX2    59-I.IIP                                     FCC0881      98     A
MCG                MI     X2,SXB1     IF II[LIS]                       FCC0881      99     A
MCG                SA4    A2-B1       DIS = LIS - 1                    FCC0881     100     A
MCG                LX4    59-D.STP                                     FCC0881     101     A
MCG                PL     X4,SXB1     IF ^ST[DIS]                      FCC0881     102     A
MCG                LX4    D.STP-D.L2P                                  FCC0881     103     A
MCG                MI     X4,SXB1     IF L2[DIS]                       FCC0881     104     A
MCG                                                                    FCC0881     105          I
          -FCC1126
MCG                SA4    A4-2        R1S = DIS - 2                    FCC0881     106          I
          -FCC1126
MCG                SA4    A4-2        R1S = DIS - 2                    FCC1126       8     A
MCG                BX6    "RN"X4                                       FCC1126       9     A
MCG                BX7    X5-X6                                        FCC1126      10     A
MCG                NZ     X7,SXB1     IF RI[R1S] " R0  */ NOT RI USE   FCC1126      11     A
MCG                                                                    FCC1126      12     A
MCG                BX6    X0*X4                                        FCC0881     107     A
MCG                BX7    X6+X1       RI[R1S] = L.TXT  */ CHANGE *RI*  FCC0881     108     A
MCG                SA7    A4                                           FCC0881     109     A
MCG                LX3    -U.PREDP                                     FCC0881     110     A
MCG                BX2    X0*X3                                        FCC0881     111     A
MCG                LX3    U.PREDP                                      FCC0881     112     A
MCG                BX6    X2+X1       PRED[UI] = L.TXT                 FCC0881     113     A
MCG                LX6    U.PREDP                                      FCC0881     114     A
MCG                SB3    B3+B1       UST = UST + 1                    FCC0881     115     A
MCG                SA6    A3                                           FCC0881     116     A
MCG                NZ     B5,SXB1     IF FSU " 0                       FCC0881     117     A
MCG                SB5    A2-"TB"     FSU = SUCC[UI]                   FCC0881     118     A
MCG                EQ     SXB1                                         FCC0881     119     A
MCG                                                                    FCC0881     120     A
MCG       *        UPDATE LINK WORD OF *R0*, FORM *XMT LT,R0* AT END OF *TXT*.     FCC0881     121     A
MCG                                                                    FCC0881     122     A
MCG       SXB3     SX5    B5                                           FCC0881     123     A
MCG                PX5    B3,X5                                        FCC0881     124     A
MCG                ALLOC  TXT,4       ALLOC( TXT , 4 )                 FCC0881     125     A
MCG                UX6    B3,X5                                        FCC0881     126     A
MCG                ALLOC  TREE,B3     ALLOC( TREE , UST );  LTR = OLD(L.TREE)     FCC0881     127     A
MCG                RJ     AIP         ADJUST POINTERS                  FCC0881     128     A
MCG                SA1    R0                                           FCC0881     129     A
MCG                SA3    "TB"+X1     LI = BASE + R0                   FCC0881     130     A
MCG                UX5    B2,X5                                        FCC0881     131     A
MCG                SB5    X2                       (B5) = O.TREE       FCC0881     132     A
MCG                SX6    B2-B1                                        FCC0881     133     A
MCG                LX6    I.USESP                                      FCC0881     134     A
MCG                SA4    A3-B1                                        FCC0881     135     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                IX6    X3-X6       USES[LI] = USES[LI] + 1 - UST                    FCC0881    136    A
MCG                SA6    A3                                                           FCC0881    137    A
MCG                SX7    B1                                                           FCC0881    138    A
MCG                LX7    D.PSP                                                        FCC0881    139    A
MCG                BX7    -X7*X4      PS[LI-1] = 0                                     FCC0881    140    A
MCG                SA7    A4                                                           FCC0881    141    A
MCG                                                                                    FCC0881    142    A
MCG                SA2    L.TXT                                                        FCC0881    143    A
MCG                SX2    X2-4        LT = L.TXT - 4                                   FCC0881    144    A
MCG                SX7    B2                                                           FCC0881    145    A
MCG                LX7    I.USESP                                                      FCC0881    146    A
MCG                SX6    B6                                                           FCC0881    147    A
MCG                LX6    I.INDXP                                                      FCC0881    148    A
MCG                BX7    X6+X7                                                        FCC0881    149    A
MCG                LX5    I.LSTP-1                                                     FCC0881    150    A
MCG                BX7    X5+X7                                                        FCC0881    151    A
MCG                SX6    B1                                                           FCC0881    152    A
MCG                BX7    X6+X7       LIT = BASE + LT                                  FCC0881    153    A
MCG                SA7    "TB"+X2     [LIT] = LW(0,UST,LTR,FSU/2,1)                    FCC0881    154    A
MCG                SA4    F.RDT+OC.XMT                                                 FCC0881    155    A
MCG                LX6    D.PSP                                                        FCC0881    156    A
MCG                BX7    X4+X6                                                        FCC0881    157    A
MCG                SA7    A7-B1       DIT = LIT - 1;  [DIT] = RDT(OC.XMT) ! PS         FCC0881    158    A
MCG                MX6    0                                                            FCC0881    159    A
MCG                SA6    A7-B1       R2T = DIT - 1;  [R2T] = 0                        FCC0881    160    A
MCG                SB3    OC.XMT                                                       FCC0881    161    A
MCG                SX1    X1                                                           FCC0881    162    A
MCG                LX1    R1.RJP                                                       FCC0881    163    A
MCG                BX7    X1+X2                                                        FCC0881    164    A
MCG                PX6    B3,X7                                                        FCC0881    165    A
MCG                SA6    A6-B1       R1T = R2T - 1;  [R1T] = TYI(OC.XMT,R0,0,LT)      FCC0881    166    A
MCG                                                                                    FCC0881    167    A
MCG      *         RESCAN *SIT* FOR *R0* AND SUBDIVIDE IT INTO 2 LISTS.               FCC0881    168    A
MCG                                                                                    FCC0881    169    A
MCG                LX3    -I.INDXP                                                     FCC0881    170    A
MCG                MX7    -I.INDXL                                                     FCC0881    171    A
MCG                BX6    -X7*X3                                                       FCC0881    172    A
MCG                SA3    B5+X6       UI = TREE + INDX[LI]                             FCC0881    173    A
MCG                SB3    A3          UO = UI                                          FCC0881    174    A
MCG                SB2    B5+B6       UX = TREE + LTR                                  FCC0881    175    A
MCG                SB4    B0          XI = 0        */ XMT LINK INSERTED               FCC0881    176    A
MCG                LX5    U.LSTP-I.LSTP                                                FCC0881    177    A
MCG                LX1    -R1.RJP                                                      FCC0881    178    A
MCG                SB5    60-U.PREDP                                                   FCC0881    179    A
MCG                EQ     SXB6                                                         FCC0881    180    A
MCG                                                                      A             FCC0881    181    A
MCG       SXB4     BX6    X3                                                           FCC0881    182    A
MCG                SA6    B3          [UO] = [UI]                                      FCC0881    183    A
MCG                SB3    B3+1        UO = UO + 1                                      FCC0881    184    A
MCG                                                                                    FCC0881    185    A
MCG       SXB5     LX3    59-U.LASTP                                                   FCC0881    186    A
MCG                MI     X3,SXB7     IF LAST[UI]                                      FCC0881    187    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SA3    A3+1        UI = UI + 1                                  FCC0881     188     A
MCG                                                                                FCC0881     189     A
MCG        SXB6    LX3    -U.SUCCP                                                 FCC0881     190     A
MCG                SA4    "TB"+X3     LIS = TB + SUCC[UI]                          FCC0881     191     A
MCG                LX3    U.SUCCP                                                  FCC0881     192     A
MCG                LX4    59-I.IIP                                                 FCC0881     193     A
MCG                MI     X4,SXB5     IF II[LI]                                    FCC0881     194     A
MCG                LX7    B5,X3                                                    FCC0881     195     A
MCG                BX6    X7-X1                                                    FCC0881     196     A
MCG                BX7    "RN"X6                                                   FCC0881     197     A
MCG                ZR     X7,SXB4     IF PRED[UI] = R0                             FCC0881     198     A
MCG                                                                                FCC0881     199     A
MCG       *        ADD EDGE TO *XMT* LIST                                          FCC0881     200     A
MCG                                                                                FCC0881     201     A
MCG                BX6    X3                                                       FCC0881     202     A
MCG                SA6    B2          [UX] = [UI]                                  FCC0881     203     A
MCG                SB2    B2+B1       UX = UX + 1                                  FCC0881     204     A
MCG                NZ     B4,SXB5     IF XI " 0     */ ^ FIRST *ST*                FCC0881     205     A
MCG                SB4    B1          XI = 1                                       FCC0881     206     A
MCG                BX6    X5+X2                                                    FCC0881     207     A
MCG                LX1    U.PREDP                                                  FCC0881     208     A
MCG                BX6    X1+X6                                                    FCC0881     209     A
MCG                LX1    -U.PREDP                                                 FCC0881     210     A
MCG                SA6    B3          [UO] = SIT(0,R0,FST,LT)                      FCC0881     211     A
MCG                SB3    B3+B1       UO = UO + 1                                  FCC0881     212     A
MCG                EQ     SXB5                                                     FCC0881     213     A
MCG                                                                                FCC0881     214     A
MCG        SXB7    SA1    B2-B1                                                    FCC0881     215     A
MCG                SA3    B3-B1                                                    FCC0881     216     A
MCG                SA4    SXBB                                                     FCC0881     217     A
MCG                MX5    1                                                        FCC0881     218     A
MCG                LX5    1+U.LASTP                                                FCC0881     219     A
MCG                BX6    X1+X5       LAST[UX-1] = 1  */ TERMINATE LISTS           FCC0881     220     A
MCG                SA6    A1                                                       FCC0881     221     A
MCG                BX7    X3+X5       LAST[UO-1] = 1                               FCC0881     222     A
MCG                SA7    A3                                                       FCC0881     223     A
MCG                NZ     X4,SXB      IF CALLF " 0  */ NOT CALL FROM *CRA*         FCC0881     224     A
MCG                SX1    X2                                                       FCC0881     225     A
MCG                SA2    "TB"+X1                                                  FCC0881     226     A
MCG                SX3    B1                                                       FCC0881     227     A
MCG                LX3    I.ICP                                                    FCC0881     228     A
MCG                BX6    X3+X2       IC[BASE+LT] = 1                              FCC0881     229     A
MCG                SA6    A2                                                       FCC0881     230     A
MCG                RJ     RIL         PUT GENERATED XMT ON *ICL*                   FCC0881     231     A
MCG                EQ     SXB                                                      FCC0881     232     A
MCG                                                                                MCG        3683     A
MCG        SXBA    BFMW   I,(USES,INDX)                                            MCG        3684          I
          -FCC0881
MCG        SXBA    BFMW   I,(USES,INDX,LST)                                        FCC0881     233     A
MCG        SXBB    BSS    1                                                        FCC0881     234     A
MCG        OXI     TITLE  OXI - OUTPUT TRANSMIT INSTRUCTION                        MCG        3685     A
MCG       **       OXI - OUTPUT XMIT INSTRUCTION ( *XMT* OR *SA* )                 MCG        3686     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG      *                                                                  MCG      3687     A
MCG      *         ENTRY  (B3) = REGR ( RESULT REGISTER )                   MCG      3688     A
MCG      *                (B4) = REGF ( FROM )                              MCG      3689     A
MCG      *                (B2) = OC , INSTRUCTION OPCODE                    MCG      3690     A
MCG      *                                                                  MCG      3691     A
MCG      *         EXIT   INSTRUCTION ISSUED, AXR AND RVT UPDATED            MCG      3692     A
MCG                                                                         MCG      3693     A
MCG       OXI      ROUTINE                                                  MCG      3694     A
MCG                SA1    RVT+B4                                            MCG      3695     A
MCG                MX7    0                                                 MCG      3696     A
MCG                BX6    X1          RVT(REGR) = RVT(REGF)                 MCG      3697     A
MCG                SA6    RVT+B3                                            MCG      3698     A
MCG                SA2    AXR                                               MCG      3699     A
MCG                SA3    RBV+B4                                            MCG      3700     A
MCG                SA4    RBV+B3                                            MCG      3701     A
MCG                SA7    A1          RVT(REGF) = 0                         MCG      3702     A
MCG                BX5    -X4*X2                                            MCG      3703     A
MCG                BX7    X5+X3       AXR = AXR & ^RBV(REGR) ! RBV(REGF)    MCG      3704     A
MCG                SA7    A2                                                MCG      3705     A
MCG                SA1    T                                                 MCG      3706     A
MCG                SA2    TRA+B3                                            MCG      3707     A
MCG      +         IX3    X1-X2                                             MCG      3708     A
MCG                PL     X3,*+1                                            MCG      3709     A
MCG                BX1    X2                                                MCG      3710     A
MCG                SX7    X1          T = MAX( T , TRA(REGR) )              MCG      3711     A
MCG                SA7    A1                                                MCG      3712     A
MCG                                                                         MCG      3713     A
MCG                SA2    F.RDT+B2                                          MCG      3714     A
MCG                BX5    "RN"X6      R = RI[RVT(REGR)]                     MCG      3715     A
MCG                LX5    R1.RJP                                            MCG      3716     A
MCG                PX7    B2,X5                                             MCG      3717     A
MCG                MX6    0                                                 MCG      3718     A
MCG                SA7    DRI         [DRI] = TYI(OC,R,0,0)                 MCG      3719     A
MCG                SA6    A7+B1       [DRI+1] = 0                           MCG      3720     A
MCG                BX7    X2                                                MCG      3721     A
MCG                SA7    A6+B1       [DRI+2] = RDT(OC)                     MCG      3722     A
MCG                SA4    A7                                                MCG      3723     A
MCG                RJ     OEI         OUTPUT THE INSTRUCTION                MCG      3724     A
MCG                                                                         MCG      3725     A
MCG      *         ADJUST THE *REG* FIELD OF THE OPERAND                    MCG      3726     A
MCG                                                                         MCG      3727     A
MCG                SA1    RVT+B3                                            MCG      3728     A
MCG                MX7    -I.REGL                                           MCG      3729     A
MCG                SA2    "TB"+X1     L = BASE + RI[RVT(REGR)]              MCG      3730     A
MCG                SX4    B3                                                MCG      3731     A
MCG                BX3    X7*X2       REG[L] = REGR                         MCG      3732     A
MCG                IX6    X3+X4                                             MCG      3733     A
MCG                SA6    A2                                                MCG      3734     A
MCG                EQ     OXI                                               MCG      3735     A
MCG       RIL      TITLE  RIL - RESET ISSUE LIST                            MCG      3736     A
MCG      **        RIL - RESET ISSUE LIST TO REMOVE ISSUED INSTRUCTION FROM MCG      3737     A
MCG      *                REGISTERS AND PLACE IT BACK ON THE ISSUE LIST.   MCG      3738     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG      *                                                                  MCG        3739    A
MCG      *         ENTRY  (X1) = R , *TXT* INDEX OF THE INSTRUCTION         MCG        3740    A
MCG      *                                                                  MCG        3741    A
MCG      *         EXIT   (B4) = LST[R] , NEW *LST* OF THE INSTRUCTION      MCG        3742    A
MCG      *                (B3) = R , *TXT* INDEX OF THE INSTRUCTION         MCG        3743    A
MCG      *                (A0) = PI , *TXT* INDEX OF PREVIOUS INSTRUCTION ON *ICL*   MCG   3744    A
MCG      *                (SIW) = *SIT* INDEX OF AN AVAIL WORD, 0 IF NONE.  MCG        3745    A
MCG      *                                                                  MCG        3746    A
MCG      *         PRESERVES - X0, B7                                       MCG        3747    A
MCG                                                                         MCG        3748    A
MCG       RIL      ROUTINE                                                  MCG        3749    A
MCG                SA5    "TB"+X1     LI = BASE + R                         MCG        3750    A
MCG                SA2    O.TREE                                            MCG        3751    A
MCG                MX7    -I.INDXL                                          MCG        3752    A
MCG                SB3    X1                                                MCG        3753    A
MCG                SB2    X2                                                MCG        3754    A
MCG                LX5    -I.INDXP                                          MCG        3755    A
MCG                BX4    -X7*X5                                            MCG        3756    A
MCG                SA3    B2+X4       UI = O.TREE + INDX[LI]                MCG        3757    A
MCG                LX5    I.INDXP-I.LSTP                                    MCG        3758    A
MCG                SB4    A3          US = UI                               MCG        3759    A
MCG                MX4    -1                                                MCG        3760    A
MCG                SB5    A3          UF = UI                               MCG        3761    A
MCG                BX5    X7*X5                                             MCG        3762    A
MCG                SX1    B1                                                MCG        3763    A
MCG                SB6    B0          N = 0                                 MCG        3764    A
MCG                LX1    I.ICP                                             MCG        3765    A
MCG                EQ     RIL3                                              MCG        3766    A
MCG                                                                         MCG        3767    A
MCG      *         REMOVE ISSUED SUCCESSORS OF R FROM *SIT*, ADJUST PRED COUNTS   MCG   3768    A
MCG      *         OF UNISSUED SUCC.                                        MCG        3769    A
MCG                                                                         MCG        3770    A
MCG       RIL1     IX7    X2-X4       NPRED[SI] = NPRED[SI] + 1             MCG        3771    A
MCG                SA7    A2                                                MCG        3772    A
MCG                                                                         MCG        3773    A
MCG       RIL2     LX3    59-U.LASTP                                        MCG        3774    A
MCG                MI     X3,RIL4     IF LAST[UI]                           MCG        3775    A
MCG                SA3    A3+1        UI = UI + 1                           MCG        3776    A
MCG                                                                         MCG        3777    A
MCG       RIL3     LX3    -U.SUCCP                                          MCG        3778    A
MCG                SA2    "TB"+X3     SI = BASE + SUCC[UI]                  MCG        3779    A
MCG                BX7    X1*X2                                             MCG        3780    A
MCG                LX3    U.SUCCP                                           MCG        3781    A
MCG                LX2    59-I.IIP                                          MCG        3782    A
MCG                MI     X2,RIL2     IF II[SI]     */ IF ISSUED            MCG        3783    A
MCG                LX6    B1,X3                                             FCC0448     151    A
MCG                ERRNZ  58-U.RDLP                                         FCC0448     152    A
MCG                MI     X6,RIL2     IF RDL[UI]    */ REMOVE REDEFINITION LINKS   FCC0448  153    A
MCG                BX6    X3                                                MCG        3784    A
MCG                SA6    B4          [US] = [UI]                           MCG        3785    A
MCG                SB4    B4+1        US = US + 1                           MCG        3786    A
MCG                ZR     X7,RIL1     IF ^IC[SI]                            MCG        3787    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG              BX6    -X1*X2      IC[SI] = 0                                    MCG    3788    A
MCG              SB6    B6+B1       N = N + 1     */ COUNT NUM OF CHANGED IC"S    MCG    3789    A
MCG              SA6    A2                                                        MCG    3790    A
MCG              EQ     RIL2                                                      MCG    3791    A
MCG       RIL    SPACE  1,10                                                      MCG    3792    A
MCG      *       FINISH RESET OF *SIT*                                            MCG    3793    A
MCG                                                                               MCG    3794    A
MCG       RIL4   SA2    B5                                                        MCG    3795    A
MCG              SB5    A3+B1       UL = UI + 1                                   MCG    3796    A
MCG              SA3    B4-B1                                                     MCG    3797    A
MCG              MX4    1                                                         MCG    3798    A
MCG              LX4    1+U.LASTP                                                 MCG    3799    A
MCG              BX6    X4+X3       LAST[US-1] = 1                                MCG    3800    A
MCG              SA6    A3                                                        MCG    3801    A
MCG              LX2    -U.LSTP                                                   MCG    3802    A
MCG              SX1    X2          RN = LST[UF]                                  MCG    3803    A
MCG              BX7    X5+X1       LST[LI] = RN  */ NEW LATE START TIME          MCG    3804    A
MCG              LX7    I.LSTP                                                    MCG    3805    A
MCG              SA7    A5                                                        MCG    3806    A
MCG                                                                               MCG    3807    A
MCG      +       SX6    B4-B2                                                     MCG    3808    A
MCG              LT     B4,B5,*+1   SIW = IF( US < UL ) THEN US-O.TREE; ELSE 0    MCG    3809    A
MCG              MX6    0                                                         MCG    3810    A
MCG              SA6    SIW                                                       MCG    3811    A
MCG              SB4    X1+                                                       MCG    3812    A
MCG              ZR     B6,RIL8     IF N = 0                                      MCG    3813    A
MCG                                                                               MCG    3814    A
MCG      *       REMOVE INSTRUCTIONS THAT ARE NO LONGER *IC"S* FROM *ICL*         MCG    3815    A
MCG                                                                               MCG    3816    A
MCG              SA3    ICL         CI = ICL                                      MCG    3817    A
MCG              MX1    -I.LINKL                                                  MCG    3818    A
MCG              S"PI"  B0          PI = 0                                        MCG    3819    A
MCG              LX4    I.ICP-U.LASTP                                             MCG    3820    A
MCG              EQ     RIL6                                                      MCG    3821    A
MCG                                                                               MCG    3822    A
MCG       RIL5   S"PI"  X3          PI = CI                                       MCG    3823    A
MCG              SX3    X5          CI = LINK[L]                                  MCG    3824    A
MCG              ZR     X3,RIL7     IF CI = 0                                     MCG    3825    A
MCG                                                                               MCG    3826    A
MCG       RIL6   SA5    "TB"+X3     L = BASE + CI                                 MCG    3827    A
MCG              BX2    X4*X5                                                     MCG    3828    A
MCG              NZ     X2,RIL5     IF IC[L]                                      MCG    3829    A
MCG                                                                               MCG    3830    A
MCG              SA2    "TB"+"PI"                                                 MCG    3831    A
MCG              BX3    -X1*X5      CI = LINK[L]                                  MCG    3832    A
MCG              SB6    B6-B1       N = N - 1                                     MCG    3833    A
MCG              BX6    X1*X2                                                     MCG    3834    A
MCG              IX7    X6+X3       LINK[BASE+PI] = CI                            MCG    3835    A
MCG              MX6    -1                                                        MCG    3836    A
MCG              SA7    A2                                                        MCG    3837    A
MCG              BX5    X1*X5       LINK[L] = 0                                   MCG    3838    A
MCG              IX6    X5-X6       NPRED[L] = 1                                  MCG    3839    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SA6    A5                                                    MCG        3840    A
MCG                NZ     B6,RIL6     IF N " 0      */ NOT END OF DELETES       MCG        3841    A
MCG                                                                             MCG        3842    A
MCG                NZ     X3,RIL7     IF CI " 0                                 MCG        3843    A
MCG                SX3    "PI"                                                  MCG        3844    A
MCG                NZ     X3,RIL7     IF PI " 0     */ ICL NOT EMPTY            MCG        3845    A
MCG                SA5    "TB"+B3     L = BASE + R                              MCG        3846    A
MCG                MX1    -I.LINKL                                              MCG        3847    A
MCG                BX6    X1*X5       LINK[L] = 0                               MCG        3848    A
MCG                SX7    B3                                                    MCG        3849    A
MCG                SA6    A5                                                    MCG        3850    A
MCG                SA7    ICL         ICL = R                                   MCG        3851    A
MCG                EQ     RIL                                                   MCG        3852    A
MCG                                                                             MCG        3853    A
MCG       RIL7     SA2    "TB"                                                  MCG        3854    A
MCG                SX6    X2                                                    MCG        3855    A
MCG                ZR     X6,RIL8     IF LINK[BASE] = 0                         MCG        3856    A
MCG                SA6    ICL         ICL = LINK[BASE]                          MCG        3857    A
MCG                BX7    X1*X2       LINK[BASE] = 0                            MCG        3858    A
MCG                SA7    A2                                                    MCG        3859    A
MCG                                                                             MCG        3860    A
MCG      *         ADD THE REMOVED INSTRUCTION BACK ONTO *ICL*                  MCG        3861    A
MCG                                                                             MCG        3862    A
MCG       RIL8     SA3    ICL         CI = ICL                                  MCG        3863    A
MCG                SX2    B4                                                    MCG        3864    A
MCG                S"PI"  B0          PI = 0                                    MCG        3865    A
MCG                LX2    I.LSTP                                                MCG        3866    A
MCG                MX1    -I.LSTL                                               MCG        3867    A
MCG                SA5    "TB"+X3     L = BASE + CI                             MCG        3868    A
MCG                LX1    I.LSTP                                                MCG        3869    A
MCG                                                                             MCG        3870    A
MCG       RIL9     BX4    -X1*X5                                                MCG        3871    A
MCG                IX6    X2-X4                                                 MCG        3872    A
MCG                MI     X6,RIL10    IF LST[LI] > RN                           MCG        3873    A
MCG                S"PI"  X3          PI = CI                                   MCG        3874    A
MCG                SX3    X5          CI = LINK[L]                              MCG        3875    A
MCG                SA5    "TB"+X5     L = BASE + CI                             MCG        3876    A
MCG                NZ     X3,RIL9     IF CI " 0                                 MCG        3877    A
MCG                                                                             MCG        3878    A
MCG       RIL10    SA5    "TB"+B3     L = BASE + R                              MCG        3879    A
MCG                MX1    -I.LINKL                                              MCG        3880    A
MCG                SB6    "PI"                                                  MCG        3881    A
MCG                BX4    X1*X5                                                 MCG        3882    A
MCG                ZR     B6,RIL11    IF PI = 0                                 MCG        3883    A
MCG                SA2    "TB"+"PI"                                             MCG        3884    A
MCG                IX6    X4+X3       LINK[L] = CI                              MCG        3885    A
MCG                SX3    B3                                                    MCG        3886    A
MCG                SA6    A5                                                    MCG        3887    A
MCG                BX1    X1*X2                                                 MCG        3888    A
MCG                IX7    X1+X3       LINK[BASE+PI] = R                         MCG        3889    A
MCG                SA7    A2                                                    MCG        3890    A
MCG                EQ     RIL                                                   MCG        3891    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                                                                                  MCG        3892    A
MCG       RIL11    SA3    ICL                                                        MCG        3893    A
MCG                IX6    X4+X3       LINK[L] = ICL                                  MCG        3894    A
MCG                SA6    A5                                                         MCG        3895    A
MCG                SX7    B3          ICL = R                                        MCG        3896    A
MCG                SA7    A3                                                         MCG        3897    A
MCG                EQ     RIL                                                        MCG        3898    A
MCG       SMB      TITLE  SMB - SET MULTIPLE USE COMPUTATION BIT                     MCG        3899    A
MCG      **        SMB - SET *MUC* BIT FOR MULTIPLE USE COMPUTATIONS WHOSE           MCG        3900    A
MCG      *         USES ARE SPREAD APART SO THAT *DRR* WILL TRY TO PUT THEM IN       MCG        3901    A
MCG      *         STORE REGISTERS AND AVOID HAVING TO OUTPUT *XMT"S* WHEN THEY      MCG        3902    A
MCG      *         ARE SAVED IN MEMORY LATER.                                        MCG        3903    A
MCG                                                                                  MCG        3904    A
MCG       SMB0     MX0    -R1.RIL+1                                                  MCG        3905          I
         -CCG11
MCG                LX0    1                                                          MCG        3906          I
         -CCG11
MCG       SMB0     SX0    RN.MASK                                                    CCG11       129    A
MCG                                                                                  MCG        3907    A
MCG       SMB      ROUTINE                                                           MCG        3908    A
MCG                SA2    L.TXT                                                      MCG        3909    A
MCG                SA5    O.TREE                                                     MCG        3910    A
MCG                MX0    -D.USESL                                                   MCG        3911    A
MCG                LX0    D.USESP                                                    MCG        3912    A
MCG                SB4    4                                                          MCG        3913    A
MCG                MX1    -I.INDXL                                                   MCG        3914    A
MCG                SB2    X2-4        I = L.TXT - 4                                  MCG        3915    A
MCG                SB3    X5                                                         MCG        3916    A
MCG                SX7    B1                                                         MCG        3917    A
MCG                SA3    SMBA                                                       MCG        3918    A
MCG                LX7    D.MUCP                                                     MCG        3919    A
MCG                S"DI"  "TB"-B1     DI = BASE - 1                                  MCG        3920    A
MCG                EQ     SMB2                                                       MCG        3921    A
MCG                                                                                  MCG        3922    A
MCG       SMB1     BX6    X7+X4       MUC[DI] = 1                                    MCG        3923    A
MCG                SA6    A4+                                                        MCG        3924    A
MCG                                                                                  MCG        3925    A
MCG       SMB2     SB2    B2-B4       I = I - 4                                      MCG        3926    A
MCG                S"DI"  "DI"+B4     DI = DI + 4                                    MCG        3927    A
MCG                ZR     B2,SMB0     IF I = 0                                       MCG        3928    A
MCG                BX2    -X0*X4                                                     MCG        3929    A
MCG                AX2    D.USESP+1                                                  MCG        3930    A
MCG                ZR     X2,SMB2     IF USES[DI] < 2                                MCG        3931    A
MCG                                                                                  MCG        3932    A
MCG                BX6    X3*X4                                                      MCG        3933    A
MCG       #DAL     IFEQ   .DAL,0                                                     MCG        3934    A
MCG                NZ     X6,SMB2     IF BITS[DI] " 0                                MCG        3935    A
MCG       #DAL     ELSE                                                              MCG        3936    A
MCG                ZR     X6,SMB3     IF BITS[DI] = 0                                MCG        3937    A
MCG                LX6    D.L2P-D.LDP                                                MCG        3938    A
MCG                BX2    X4*X6                                                      MCG        3939    A
MCG                LX2    59-D.L2P                                                   MCG        3940    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                PL     X2,SMB2     IF ^( L2[DI]&LD[DI] )                    MCG        3941     A
MCG        SMB3    BSS    0                                                    MCG        3942     A
MCG        #DAL    ENDIF                                                       MCG        3943     A
MCG                                                                            MCG        3944     A
MCG       *        CHECK DISTANCE BETWEEN USES                                 MCG        3945     A
MCG                                                                            MCG        3946     A
MCG                SA2    "DI"+B1     LI = DI + 1                              MCG        3947     A
MCG                LX2    -I.INDXP                                            MCG        3948     A
MCG                BX6    -X1*X2                                              MCG        3949     A
MCG                SA5    B3+X6       UI = O.TREE + INDX[LI]                  MCG        3950     A
MCG                LX5    -U.SUCCP                                            MCG        3951     A
MCG                SB5    X5+MIN.D    SL = SUCC[UI] + MIN.D                   MCG        3952     A
MCG                LX5    U.SUCCP+59-U.LASTP                                  MCG        3953     A
MCG                MI     X5,SMB2     IF LAST[UI]   */ 2 USES IN 1 INSTRUCTION MCG        3954     A
MCG                                                                            MCG        3955     A
MCG        SMB4    SA5    A5+B1       UI = UI + 1                             MCG        3956     A
MCG                LX5    -U.SUCCP                                            MCG        3957     A
MCG                SB6    X5          SI = SUCC[UI]                           MCG        3958     A
MCG                GT     B6,B5,SMB1  IF SI > SL    */ DIST BETWEEN USES IS BIG MCG      3959     A
MCG                LX5    U.SUCCP+59-U.LASTP                                  MCG        3960     A
MCG                SB5    B6+MIN.D    SL = SI + MIN.D                         MCG        3961     A
MCG                PL     X5,SMB4     IF ^LAST[UI]                            MCG        3962     A
MCG                EQ     SMB2                                                MCG        3963     A
MCG                                                                            MCG        3964     A
MCG        SMBA    BFMW   D,(RJRS,PS,PRS,ECJ,LD,ST,JP,XMT,KS,PI,ZP)           MCG        3965     A
MCG        SNU     TITLE  SNU - SET NEXT USE DISTANCE                         MCG        3966     A
MCG       **       SNU - SET NEXT USE OF OPERANDS OF INSTRUCTION IN *RVT*     MCG        3967     A
MCG       *        NOTE - *SNU* MAY ONLY BE CALLED AFTER A CALL TO *DUC*.     MCG        3968     A
MCG       *                                                                   MCG        3969     A
MCG       *        ENTRY  (A4,X4) = DI , [DI]                                 MCG        3970     A
MCG       *                                                                   MCG        3971     A
MCG       *        EXIT   *NU* FIELD UPDATED IN *RVT* OF OPERANDS OF INSTRUCTION MCG     3972     A
MCG       *               STORES ADDED TO ISSUED STORE CHAIN ( *ISC* )        MCG        3973     A
MCG       *               (X7) = UOR = LIST OF UNCLEARED OPERAND REGISTERS    MCG        3974     A
MCG       *                                                                   MCG        3975     A
MCG       *        USES   B - 2, 6 , X - 1, 2, 3, 4, 5, 6, 7                  MCG        3976     A
MCG                                                                            MCG        3977     A
MCG        SNU0    MX7    0           UOR = 0                                 MCG        3978     A
MCG                MI     X4,SNU      IF TYPE[DI] = IV                        MCG        3979     A
MCG                LX4    1+D.TYP-D.PIP                                       MCG        3980     A
MCG                PL     X4,SNU6     IF ^PI[DI]    */ NOT A PSEUDO INSTRUCTION MCG      3981     A
MCG                SA3    "DI"-2      R1 = DI - 2                             MCG        3982     A
MCG                BX5    "RN"X3      R = RNUMS = RI[R1]                      MCG        3983     A
MCG                UX6    B2,X3                                               MCG        3984     A
MCG                SX7    B2-OC.DEF                                           MCG        3985     A
MCG                NZ     X7,SNU2     IF OC[R1] " OC.DEF                      MCG        3986     A
MCG                                                                            MCG        3987     A
MCG       *        SEARCH *UIT* AND REMOVE LOGICAL LINKS FROM THE SUCCESSOR LIST MCG     3988     A
MCG                                                                            MCG        3989     A
MCG                SA2    "DI"+B1     LI = DI + 1                             MCG        3990     A
MCG                LX4    1+D.PIP                                             FCC0393      16     A
MCG                SA3    O.TREE                                              MCG        3991     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                MX7    -I.INDXL                                                MCG        3992    A
MCG                LX2    -I.INDXP                                                MCG        3993    A
MCG                SB6    X3                                                      MCG        3994    A
MCG                BX6    -X7*X2                                                  MCG        3995    A
MCG                SA2    B6+X6       UI = O.TREE + INDX[LI]                      MCG        3996    A
MCG                SA3    A2-B1                                                   MCG        3997    A
MCG                SB2    A2          FLAG = UI     */ 0 IF SOMETHING STORED      MCG        3998    A
MCG                SB6    "TB"-B1     DB = BASE - 1                               MCG        3999    A
MCG                BX6    X3                                                      MCG        4000    A
MCG                SA6    A3          US = UI - 1;  [US] = [UI-1]                 MCG        4001    A
MCG                                                                               MCG        4002    A
MCG       RLL1     LX2    -U.SUCCP                                                MCG        4003    A
MCG                SA3    B6+X2       DJ = DB + SUCC[UI]                          MCG        4004    A
MCG                LX2    U.SUCCP                                                 MCG        4005    A
MCG                LX3    58-D.TYP                                                MCG        4006    A
MCG                LX6    B1,X3                                                   MCG        4007    A
MCG                MI     X6,RLL4     IF TYPE[DJ] = II ! IV                       MCG        4008    A
MCG                SA1    A3-2        R1J = DJ - 2                                MCG        4009    A
MCG                BX6    X5-X1                                                   MCG        4010    A
MCG                BX7    "RN"X6                                                  MCG        4011    A
MCG                ZR     X7,RLL3     IF RI[R1J] = R                              MCG        4012         I
          -FCC0393
MCG                MI     X3,RLL2     IF TYPE[DJ] = III                           MCG        4013         I
          -FCC0393
MCG                NZ     X7,RLL1A    IF RI[R1J] " R                              FCC0393      17    A
MCG                                                                               FCC0393      18    A
MCG       *        DEF USED IN *RI* OF ANOTHER INSTRUCTION, RESET *PS* IF A STORE FCC0393      19    A
MCG                                                                               FCC0393      20    A
MCG                LX3    1+D.TYP-D.STP                                           FCC0393      21    A
MCG                PL     X3,RLL3     IF ^ST[DJ]                                  FCC0393      22    A
MCG       #DAL     IFNE   .DAL,0                                                  FCC0393      23    A
MCG                LX3    D.STP-D.L2P                                             FCC0393      24    A
MCG                MI     X3,RLL3     IF L2[DJ]                                   FCC0393      25    A
MCG       #DAL     ENDIF                                                          FCC0393      26    A
MCG                SX3    B1                                                      FCC0393      27    A
MCG                LX3    D.PSP                                                   FCC0393      28    A
MCG                BX7    X3+X4       PS[DI] = 1    */ INDICATE THAT *DEF* IS *PS FCC0393      29    A
MCG                SA7    A4                                                      FCC0393      30    A
MCG                EQ     RLL3                                                    FCC0393      31    A
MCG                                                                               FCC0393      32    A
MCG       RLL1A    MI     X3,RLL2     IF TYPE[DJ] = III                           FCC0393      33    A
MCG                LX1    -R1.RJP                                                 MCG        4014    A
MCG                BX6    X5-X1                                                   MCG        4015    A
MCG                LX1    R1.RJP-R1.RKP                                           MCG        4016    A
MCG                BX7    "RN"X6                                                  MCG        4017    A
MCG                ZR     X7,RLL3     IF RJ[R1J] = R                              MCG        4018    A
MCG                BX6    X5-X1                                                   MCG        4019    A
MCG                BX7    "RN"X6                                                  MCG        4020    A
MCG                ZR     X7,RLL3     IF RK[R1J] = R                              MCG        4021    A
MCG                EQ     RLL4                                                    MCG        4022    A
MCG                                                                               MCG        4023    A
MCG       RLL2     SA1    A1+B1       R2J = R1J + 1                               MCG        4024    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                LX1    -IH.RFP                                          MCG        4025    A
MCG                BX6    X5-X1                                            MCG        4026    A
MCG                BX7    "RN"X6                                           MCG        4027    A
MCG                NZ     X7,RLL4     IF RF[R2J] " R                       MCG        4028    A
MCG        RLL3    BX6    X2                                               MCG        4029    A
MCG                SA6    A6+B1       US = US + 1;  [US] = [UI]            MCG        4030    A
MCG                SB2    0           FLAG = 0                             MCG        4031    A
MCG                                                                       MCG        4032    A
MCG        RLL4    BX6    X2          UL = [UI]                            MCG        4033    A
MCG                SA2    A2+B1       UI = UI + 1                          MCG        4034    A
MCG                LX6    59-U.LASTP                                       MCG        4035    A
MCG                PL     X6,RLL1     IF ^LAST[UL]                         MCG        4036    A
MCG                                                                       MCG        4037    A
MCG                MX7    1                                                MCG        4038    A
MCG                LX7    1+U.LASTP                                        MCG        4039    A
MCG                NZ     B2,RLL5     IF FLAG " 0   */ NOTHING STORED      MCG        4040    A
MCG                SA2    A6                                               MCG        4041    A
MCG                BX6    X7+X2       LAST[US] = 1                         MCG        4042    A
MCG                SA6    A2                                               MCG        4043    A
MCG                EQ     SNU2                                             MCG        4044    A
MCG                                                                       MCG        4045    A
MCG        RLL5    SA3    L.TXT                                            MCG        4046    A
MCG                SX4    X3-4                                             MCG        4047    A
MCG                BX6    X7+X4                                            MCG        4048    A
MCG                SA6    B2          [FLAG] = UIT(1,0,0,L.TXT-4)          MCG        4049    A
MCG                EQ     SNU2                                             MCG        4050    A
MCG                                                                       MCG        4051    A
MCG        SNU     ROUTINE                                                 MCG        4052    A
MCG                LX4    58-D.TYP                                         MCG        4053    A
MCG                IX7    X4+X4                                            MCG        4054    A
MCG                MI     X7,SNU0     IF TYPE[DI] = II ! IV                MCG        4055    A
MCG                PL     X4,SNU1     IF TYPE[DI] = I                      MCG        4056    A
MCG                                                                       MCG        4057    A
MCG                SA3    "DI"-B1     R2 = DI - 1                          MCG        4058    A
MCG                LX4    1+D.TYP-D.^DP                                    MCG        4059    A
MCG                LX3    -IH.RFP                                          MCG        4060    A
MCG                BX5    "RN"X3      RNUMS = RF[R2]                       MCG        4061    A
MCG                PL     X4,SNU2     IF DEF[DI]                           MCG        4062    A
MCG                SA3    A3-B1       R1 = R2 - 1                          MCG        4063    A
MCG                LX5    R1.RIL                                           MCG        4064    A
MCG                BX2    "RN"X3                                           MCG        4065    A
MCG                IX5    X2+X5       RNUMS = SHIFT( RNUMS , 16) ! RI[R1]  MCG        4066    A
MCG                EQ     SNU2                                             MCG        4067    A
MCG                                                                       MCG        4068    A
MCG        SNU1    LX4    2+D.TYP-D.TYIOP                                  MCG        4069    A
MCG                MX6    -D.TYIOL                                         MCG        4070    A
MCG                SA5    "DI"-2      R1 = DI - 2                          MCG        4071    A
MCG                BX7    -X6*X4                                           MCG        4072    A
MCG                SA3    SNUB+X7                                          MCG        4073          I
           -FCC0719
MCG                SA3    TYIOM+X7                                         FCC0719       5    A
MCG                UX2    B2,X3                                            MCG        4074    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                BX1    X2*X5                                          MCG        4075    A
MCG                AX5    B2,X1       RNUMS = TYIO[R1,DI]                MCG        4076    A
MCG                NZ     X7,SNU2     IF TYIO[DI] " 0  */ NOT RJ,RK CASE MCG        4077    A
MCG                AX6    B2,X5                                          MCG        4078    A
MCG                BX7    X6-X5                                          MCG        4079    A
MCG                BX3    "RN"X7                                         MCG        4080    A
MCG                NZ     X3,SNU2     IF RJ[R1] " RK[R1]                 MCG        4081    A
MCG                AX5    B2,X5       RNUMS = RJ[R1]                     MCG        4082    A
MCG                                                                      MCG        4083    A
MCG        SNU2    SA3    O.TREE                                         MCG        4084    A
MCG                MX1    -I.USESL                                       MCG        4085    A
MCG                SB6    X3                                             MCG        4086    A
MCG                IX7    X7-X7       UOR = 0                            MCG        4087    A
MCG                LX1    I.USESP                                        MCG        4088    A
MCG                MX4    -I.INDXL                                       MCG        4089    A
MCG                                                                      MCG        4090    A
MCG      *         SET *TXT* INDEX OF NEXT USE OF OPERANDS IN *RVT*      MCG        4091    A
MCG                                                                      MCG        4092    A
MCG        SNU3    BX6    "RN"X5      R = RI[RNUMS]                      MCG        4093    A
MCG                ZR     X6,SNU5     IF R = 0                           MCG        4094    A
MCG                SA3    "TB"+X6     L = BASE + R                       MCG        4095    A
MCG                BX2    -X1*X3                                         MCG        4096    A
MCG                ZR     X2,SNU5     IF USES[L] = 0                     MCG        4097    A
MCG                                                                      MCG        4098    A
MCG      *         SEARCH *SIT* TO FIND FIRST UNISSUED SUCCESSOR OF OPERAND MCG      4099    A
MCG                                                                      MCG        4100    A
MCG                LX3    -I.INDXP                                       MCG        4101    A
MCG                BX6    -X4*X3                                         MCG        4102    A
MCG                LX3    I.INDXP-I.REGP                                 MCG        4103    A
MCG                SA2    B6+X6       UI = O.TREE + INDX[L]              MCG        4104    A
MCG                SX6    X3          REGNO = REG[L]                     MCG        4105    A
MCG        SNU4    LX2    -U.SUCCP                                       MCG        4106    A
MCG                SA3    "TB"+X2     SI = BASE + SUCC[UI]               MCG        4107    A
MCG                LX3    59-I.IIP                                       MCG        4108    A
MCG                SA2    A2+B1       UI = UI + 1                        MCG        4109        I
         -FCC0448
MCG                MI     X3,SNU4     IF II[SI]                          MCG        4110        I
         -FCC0448
MCG                LX2    59-U.RDLP+U.SUCCP                              FCC0448     154    A
MCG                BX3    X2+X3                                          FCC0448     155    A
MCG                SA2    A2+B1       UI = UI + 1                        FCC0448     156    A
MCG                MI     X3,SNU4     IF II[SI] ! RDL[UI-1]              FCC0448     157    A
MCG                                                                      MCG        4111    A
MCG                SA2    RVT+X6                                         MCG        4112    A
MCG                LX7    18                                             MCG        4113    A
MCG                IX7    X6+X7       UOR = SHIFT( UOR , 18 ) ! REGNO    MCG        4114    A
MCG                LX2    -RV.NUP                                        MCG        4115    A
MCG                SX6    A3-"TB"                                        MCG        4116    A
MCG                BX3    X0*X2                                          MCG        4117    A
MCG                IX6    X6+X3       NU[ RVT(REGNO) ] = SUCC[UI-1]      MCG        4118    A
MCG                LX6    RV.NUP                                         MCG        4119    A
MCG                SA6    A2                                             MCG        4120    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                                                                                  MCG        4121    A
MCG       SNU5     AX5    R1.RIL      RNUMS = SHIFT( RNUMS , - 16 )                  MCG        4122    A
MCG                NZ     X5,SNU3     IF RNUMS " 0                                   MCG        4123    A
MCG                                                                                  MCG        4124    A
MCG      *         CHECK FOR STORES AND OPS THAT *PRS*, ADJUST APPROPIATE FLAGS      MCG        4125    A
MCG                                                                                  MCG        4126    A
MCG       SNU6     S"DI"  "DI"                                                       MCG        4127    A
MCG                SA1    SNUA                                                       MCG        4128    A
MCG                BX6    X1*X4                                                      MCG        4129    A
MCG                ZR     X6,SNU      IF ST[DI] & PRS[DI] = 0                        MCG        4130    A
MCG                LX4    59-D.STP                                                   MCG        4131    A
MCG                MI     X4,SNU7     IF ST[DI]                                      MCG        4132    A
MCG                                                                                  MCG        4133    A
MCG                SA5    "DI"+2      R1RS = DI + 2                                  MCG        4134    A
MCG                BX6    "RN"X5                                                     MCG        4135    A
MCG                LX5    -R1.SOP                                                    MCG        4136    A
MCG                MX1    -SO.RTL-SO.RNL                                             MCG        4137    A
MCG                BX3    -X1*X5      RN = SOREG[R1RS]                               MCG        4138    A
MCG                SX2    X3-8                                                       MCG        4139    A
MCG                PL     X2,SNU      IF RN > 7     */ IF NOT A B-REGISTER           MCG        4140    A
MCG                SA6    LRN+X3      LRN(RN) = R   */ SAVE LAST R-NUMBER IN B-REG   MCG        4141    A
MCG                EQ     SNU                                                        MCG        4142    A
MCG                                                                                  MCG        4143    A
MCG       SNU7     SA1    ISC                                                        MCG        4144    A
MCG                MX3    -I.LINKL                                                   MCG        4145    A
MCG                SA5    "DI"+B1     L = DI + 1                                     MCG        4146    A
MCG                BX2    X3*X5                                                      MCG        4147    A
MCG                BX6    X2+X1       LINK[L] = ISC */ POINT THIS TO LAST            MCG        4148    A
MCG                SA6    A5                                                         MCG        4149    A
MCG                SX6    A5-"TB"                                                    MCG        4150    A
MCG                SA6    A1          ISC = L - BASE  */ CHAIN POINTS TO THIS        MCG        4151    A
MCG                                                                                  MCG        4152    A
MCG      *         ADJUST *PS* BIT FOR THE *RI* IN THE STORE                         MCG        4153    A
MCG                                                                                  MCG        4154    A
MCG                SA1    "DI"-2      R1 = DI - 2                                    MCG        4155    A
MCG                BX6    "RN"X1      R = RI[R1]                                     MCG        4156         I
          -FCC0393
MCG                SA5    "TB"+X6     L = BASE + R                                   MCG        4157         I
          -FCC0393
MCG                BX4    "RN"X1      R = RI[R1]                                     FCC0393      34    A
MCG                SA5    "TB"+X4     L = BASE + R                                   FCC0393      35    A
MCG                MX6    -I.INDXL                                                   MCG        4158    A
MCG                SA3    O.TREE                                                     MCG        4159    A
MCG                LX5    -I.INDXP                                                   MCG        4160    A
MCG                SB2    X3                                                         MCG        4161    A
MCG                BX2    -X6*X5                                                     MCG        4162    A
MCG                SA3    B2+X2       UI = O.TREE + INDX[L]                          MCG        4163    A
MCG                                                                                  MCG        4164    A
MCG       SNU9     LX3    -U.SUCCP                                                   MCG        4165    A
MCG                SA2    "TB"+X3     SI = BASE + SUCC[UI]                           MCG        4166    A
MCG                LX2    59-I.IIP                                                   MCG        4167    A
MCG                MI     X2,SNU10    IF II[SI]                                      MCG        4168    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SA1    A2-3        R1S = SI - 3                                FCC0393      36    A
MCG                BX2    X1-X4                                                   FCC0393      37    A
MCG                BX6    "RN"X2                                                  FCC0393      38    A
MCG                NZ     X6,SNU10    IF RI[R1S] " R                              FCC0393      39    A
MCG                SA1    A2-B1       D = SI - 1                                  MCG        4169    A
MCG                LX1    59-D.STP                                                MCG        4170    A
MCG        #DAL    IFEQ   .DAL,0                                                  MCG        4171    A
MCG                MI     X1,SNU      IF ST[DI]     */ AN UNISSUED STORE          MCG        4172    A
MCG        #DAL    ELSE                                                           MCG        4173    A
MCG                PL     X1,SNU10    IF ^ST[DI]                                  MCG        4174    A
MCG                LX1    D.STP-D.L2P                                             MCG        4175    A
MCG                PL     X1,SNU      IF ^L2[DI]                                  MCG        4176    A
MCG        #DAL    ENDIF                                                          MCG        4177    A
MCG        SNU10   LX3    U.SUCCP+59-U.LASTP                                      MCG        4178    A
MCG                BX6    X3          UL = UI                                     MCG        4179    A
MCG                SA3    A3+B1       UI = UI + 1                                 MCG        4180    A
MCG                PL     X6,SNU9     IF ^LAST[UL]                                MCG        4181    A
MCG                                                                               MCG        4182    A
MCG                SA5    A5-B1       D = L - 1                                   MCG        4183    A
MCG                SX4    B1                                                      MCG        4184    A
MCG                LX4    D.PSP                                                   MCG        4185    A
MCG                BX6    -X4*X5      PS[D] = 0     */ CLEAR *PS* BIT             MCG        4186    A
MCG                SA6    A5                                                      MCG        4187    A
MCG                EQ     SNU                                                     MCG        4188    A
MCG                                                                               MCG        4189    A
MCG        SNUA    BFMW   D,(PRS,ST)                                              MCG        4190    A
MCG        TGL     EJECT                                                          CCGA023      38    A
MCG                TITLE  TGL - TOGGLE TWO REGISTERS.                             CCGA023      39    A
MCG       **       TGL - THIS ROUTINE WILL BE INOLKED BEFORE UJR                  CCGA023      40    A
MCG       *        TRIES TO STORE OUT A REGISTER.  BASICALLY IT WILL              CCGA023      41    A
MCG       *        CHECK TO SEE IF R0 IS AN OPERAND OF T0.  IF THIS               CCGA023      42    A
MCG       *        HAPPENS R0 WILL NOT BE STORED TO MEMORY BUT WILL               CCGA023      43    A
MCG       *        BE SIMPLY TOGGLED WITH AN AVAILABLE REGISTER OF THE            CCGA023      44    A
MCG       *        CLASS SOUGHT BY T0.  THIS MODULE IS AN ANSWER TO               CCGA023      45    A
MCG       *        FCC2207 WHERE SRM OF R0 REMOVED T0 FROM THE ISSUE              CCGA023      46    A
MCG       *        CANDIDATE LIST.                                                CCGA023      47    A
MCG       *                                                                       CCGA023      48    A
MCG       *        ENTRY  (B3) = REGNO CURRENTLY ASSIGNED TO R0                   CCGA023      49    A
MCG       *               (T0) = INSTRUCTION TO BE ISSUED                         CCGA023      50    A
MCG       *               (R0) = RESULT TO BE STORED                              CCGA023      51    A
MCG       *               (CLASS) = CLASS BITS FOR T0                             CCGA023      52    A
MCG                                                                               CCGA023      53    A
MCG       *        EXIT   (B4) = REGNO. TOGGLED WITH R0 (= 0 NO TOGGLE)           CCGA023      54    A
MCG                                                                               CCGA023      55    A
MCG       *        MAINTAINS B3,T0,CLASS,R0                                       CCGA023      56    A
MCG                                                                               CCGA023      57    A
MCG        RTGL    BSS    1                                                       CCGA023      58    A
MCG                                                                               CCGA023      59    A
MCG        TGL     ROUTINE                                                        CCGA023      60    A
MCG                                                                               CCGA023      61    A
MCG       *        FIND OPERANDS OF T0.                                           CCGA023      62    A
MCG                                                                               CCGA023      63    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SB4    B0                                          CCGA023      64     A
MCG                SA2    T0                                          CCGA023      65     A
MCG                RJ     GOR         GET OPERAND ROUTINE             CCGA023      66     A
MCG                                                                   CCGA023      67     A
MCG      *         LOOK IN OPERAND LIST FOR R0.                       CCGA023      68     A
MCG                                                                   CCGA023      69     A
MCG                SX1    B3                                          CCGA023      70     A
MCG       TGL20    ZR     X7,TGL      IF ROP = 0  */EXIT              CCGA023      71     A
MCG                BX5    X1-X7                                       CCGA023      72     A
MCG                SX6    X5                                          CCGA023      73     A
MCG                ZR     X6,TGL40    IF ROP = REGR0                  CCGA023      74     A
MCG                AX7    18                                          CCGA023      75     A
MCG                EQ     TGL20                                       CCGA023      76     A
MCG                                                                   CCGA023      77     A
MCG      *         R0 IS AN OPERAND OF T0 - TOGGLE IT.                CCGA023      78     A
MCG      *         CHECK IF R0 HAS MORE THAN ONE USE - IF SO FORGET IT. CCGA023    79     A
MCG                                                                   CCGA023      80     A
MCG       TGL40    SA1    R0                                          CCGA023      81     A
MCG                BX1    "RN"X1                                      CCGA023      82     A
MCG                SA5    "TB"+X1     LT = BASE + R0                  CCGA023      83     A
MCG                MX7    -I.USESL                                    CCGA023      84     A
MCG                LX5    59-I.USESP                                  CCGA023      85     A
MCG                BX6    -X7*X5                                      CCGA023      86     A
MCG                NZ     X6,*+1S17   IF USES[LT] > 1 BLOWUP          CCGA023      87     A
MCG                                                                   CCGA023      88     A
MCG      *         FIND A REGISTER TO TOGGLE WITH R0.                 CCGA023      89     A
MCG                                                                   CCGA023      90     A
MCG                SA3    CLASS                                       CCGA023      91     A
MCG                SA2    LXR                                         CCGA023      92     A
MCG                BX7    -X2*X3      TRV = ^LXR & CLASS              CCGA023      93     A
MCG                ZR     X7,*+1S17   IF TRV = 0 BLOWUP               CCGA023      94     A
MCG                NX6    B4,X7                                       CCGA023      95     A
MCG                SA3    B4+FXRA+16-47                               CCGA023      96     A
MCG                SB4    X3          REGTGL = FIRST TRV              CCGA023      97     A
MCG                                                                   CCGA023      98     A
MCG      *         MAKE R0 UNAVAILABLE AND RESTORE RVT(REGR0)         CCGA023      99     A
MCG                                                                   CCGA023     100     A
MCG                SA2    RBV+B3                                      CCGA023     101     A
MCG                SA3    AXR                                         CCGA023     102     A
MCG                SX7    X1                                          CCGA023     103     A
MCG                BX6    X3-X2                                       CCGA023     104     A
MCG                SA7    RVT+B3      RVT(REGRO) = R0                 CCGA023     105     A
MCG                SA6    A3          MAKE REGRO UNAVAIL.             CCGA023     106     A
MCG                                                                   CCGA023     107     A
MCG      *         TOGGLE REGTGL AND R0.                              CCGA023     108     A
MCG      *         OUTPUT THE FOLLOWING SEQUENCE.                     CCGA023     109     A
MCG      *         XOR    R0,R0,RTGL                                  CCGA023     110     A
MCG      *         XOR    RTGL,R0,RTGL                                CCGA023     111     A
MCG      *         XOR    R0,R0,RTGL                                  CCGA023     112     A
MCG                                                                   CCGA023     113     A
MCG                SA2    RVT+B4                                      CCGA023     114     A
MCG                PX7    B3,X1                                       CCGA023     115     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG          BX2    "RN"X2                                          CCGA023    116    A
MCG          PX6    B4,X2                                           CCGA023    117    A
MCG          SA7    A1                                              CCGA023    118    A
MCG          SA6    RTGL                                            CCGA023    119    A
MCG          SB2    OC.XOR                                          CCGA023    120    A
MCG          SA3    =XF.RDT+B2                                      CCGA023    121    A
MCG          BX1    "RN"X1      R0                                  CCGA023    122    A
MCG          BX2    "RN"X2      RTGL                                CCGA023    123    A
MCG          LX1    R1.RJP                                          CCGA023    124    A
MCG          LX2    R1.RKP                                          CCGA023    125    A
MCG          BX7    X1+X2                                           CCGA023    126    A
MCG          PX6    B2,X7                                           CCGA023    127    A
MCG          MX7    0                                               CCGA023    128    A
MCG          SA6    DRI         DRI(0) = TYI(OC.XOR,R0,RTGL,0)      CCGA023    129    A
MCG          SA7    A6+B1       DRI(1) = 0                          CCGA023    130    A
MCG          BX6    X3                                              CCGA023    131    A
MCG          SA6    A7+B1       DRI(2) = FX.RDT(OC.XOR)             CCGA023    132    A
MCG          SA4    A6                                              CCGA023    133    A
MCG          SX7    B3                                              CCGA023    134    A
MCG          SA7    A4+B1       DRI(3) = REGR0                      CCGA023    135    A
MCG          RJ     OEI         ISSUE XOR REGR0,R0,RTGL             CCGA023    136    A
MCG          SA1    R0                                              CCGA023    137    A
MCG          SA2    RTGL                                            CCGA023    138    A
MCG          UX6    B4,X1                                           CCGA023    139    A
MCG          UX7    B3,X2                                           CCGA023    140    A
MCG          SA4    DRI+2                                           CCGA023    141    A
MCG          SX7    B3                                              CCGA023    142    A
MCG          SA7    A4+B1       DRI(3) = REGR0                      CCGA023    143    A
MCG          RJ     OEI         ISSUE XOR REGTGL,R0,RTGL            CCGA023    144    A
MCG          SA1    R0                                              CCGA023    145    A
MCG          SA2    RTGL                                            CCGA023    146    A
MCG          UX6    B3,X1                                           CCGA023    147    A
MCG          UX7    B4,X2                                           CCGA023    148    A
MCG          SA4    DRI+2                                           CCGA023    149    A
MCG          SX7    B3                                              CCGA023    150    A
MCG          SA7    A4+B1       DRI(3) = REGTGL                     CCGA023    151    A
MCG          RJ     OEI         ISSUE XOR REGR0,R0,RTGL             CCGA023    152    A
MCG                                                                CCGA023    153    A
MCG     *    TOGGLE RVT ENTRIES AND LINK WORDS.                     CCGA023    154    A
MCG                                                                CCGA023    155    A
MCG          SA1    R0                                              CCGA023    156    A
MCG          SA2    RTGL                                            CCGA023    157    A
MCG          UX3    B3,X1       (B3) = REGR0                        CCGA023    158    A
MCG          UX4    B4,X2       (B4) = REGTGL                       CCGA023    159    A
MCG          SA1    RVT+B3                                          CCGA023    160    A
MCG          SA2    RVT+B4                                          CCGA023    161    A
MCG          BX6    X1                                              CCGA023    162    A
MCG          BX7    X2                                              CCGA023    163    A
MCG          SA6    A2          RVT(REGR0) = RVT(REGTGL)            CCGA023    164    A
MCG          SA7    A1          RVT(REGTGL) = RVT(REGR0)            CCGA023    165    A
MCG          SA1    "TB"+X3     LIR0 = BASE + R0                    CCGA023    166    A
MCG          SA2    "TB"+X4     LIRTGL = BASE + RTGL                CCGA023    167    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                BX6    X2                                                  CCGA023      168     A
MCG                BX7    X1                                                  CCGA023      169     A
MCG                SA6    A1          LIR0 = LIRTGL                           CCGA023      170     A
MCG                SA7    A2          LIRTGL = LIR0                           CCGA023      171     A
MCG                                                                           CCGA023      172     A
MCG      *         RETURN                                                    CCGA023      173     A
MCG                EQ     TGL                                                 CCGA023      174     A
MCG       SNUB     VFD    12/2020B,16/-1,16/-1,16/0                           MCG         4191          I
         -FCC0719
MCG                VFD    12/2020B,16/0,16/-1,16/0                            MCG         4192          I
         -FCC0719
MCG                VFD    12/2000B,32/0,16/-1   SDS,SST , IGNORE RJ AND RK     MCG         4193          I
         -FCC0719
MCG       SRM      TITLE  SRM - SAVE RESULT IN MEMORY                         MCG         4194     A
MCG      **        SRM - SAVE RESULT IN MEMORY AND SET FOR RELOAD OF INSTRUCTION  MCG     4195     A
MCG      *                                                                    MCG         4196     A
MCG      *         ENTRY  (B3) = REGNO , OF REGISTER TO BE STORED OUT          MCG         4197     A
MCG      *                (R0) = *TXT* INDEX OF INSTRUCTION                   MCG         4198     A
MCG      *                (T0) = *TXT* INDEX OF INSTRUCTION TO ISSUE          MCG         4199     A
MCG                                                                           MCG         4200     A
MCG       SRM      ROUTINE                                                    MCG         4201     A
MCG                SA1    R0                                                  MCG         4202     A
MCG                SA2    O.TREE                                              MCG         4203     A
MCG                SA5    "TB"+X1     LI = BASE + R0                          MCG         4204     A
MCG                MX7    -I.INDXL                                            MCG         4205     A
MCG                SB2    X2                                                  MCG         4206     A
MCG                LX5    -I.INDXP                                            MCG         4207     A
MCG                BX6    -X7*X5                                              MCG         4208     A
MCG                SA4    A5-B1       DI = LI - 1                             MCG         4209     A
MCG                SB6    B2+X6       U0 = O.TREE + INDX[LI]                  MCG         4210     A
MCG                LX4    59-D.EQVP                                           MCG         4211     A
MCG                PL     X4,SRM3     IF ^EQV[DI]   */ NO EQUIV LINKS         MCG         4212     A
MCG                                                                           MCG         4213     A
MCG      *         INSTRUCTION IS A PREDECESSOR OF INTERFERING STORES, REMOVE  MCG         4214     A
MCG      *         THE LOGICAL LINKS FROM *SIT*                                MCG         4215     A
MCG                                                                           MCG         4216     A
MCG                SA3    B6          UI = U0                                 MCG         4217     A
MCG                SB5    B6          US = U0                                 MCG         4218     A
MCG                                                                           MCG         4219     A
MCG       SRM1     BX6    X3          UL = [UI]                               MCG         4220     A
MCG                LX3    59-U.EQVP                                           MCG         4221     A
MCG                MI     X3,SRM2     IF EQV[UI]                              MCG         4222     A
MCG                SA6    B5+         [US] = UL                               MCG         4223     A
MCG                SB5    B5+1        US = US + 1                             MCG         4224     A
MCG       SRM2     SA3    A3+B1       UI = UI + 1                             MCG         4225     A
MCG                LX6    59-U.LASTP                                          MCG         4226     A
MCG                PL     X6,SRM1     IF ^LAST[UL]                            MCG         4227     A
MCG                                                                           MCG         4228     A
MCG                SA3    B5-B1                                               MCG         4229     A
MCG                MX7    1                                                   MCG         4230     A
MCG                LX7    1+U.LASTP                                           MCG         4231     A
MCG                BX6    X7+X3       LAST[US-1] = 1                          MCG         4232     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                SA6    A3+                                                    MCG        4233     A
MCG                                                                               MCG        4234     A
MCG      *         SEARCH SUCCESSORS OF *R0* FOR AN ISSUED STORE TO A             MCG        4235     A
MCG      *         *CHEIP* ADDRESS.                                               MCG        4236     A
MCG                                                                               MCG        4237     A
MCG       SRM3     SA3    B6          UI = U0                                     MCG        4238     A
MCG                SA1    SRMA                                                   MCG        4239     A
MCG                R=     B6,59-U.LASTP+U.SUCCP                                  MCG        4240     A
MCG                SB2    59-D.STP                                               MCG        4241     A
MCG                SB5    IH.RFP                                                 MCG        4242     A
MCG                                                                               MCG        4243     A
MCG       SRM4     LX3    -U.SUCCP                                               MCG        4244     A
MCG                SA5    "TB"+X3     SI = BASE + SUCC[UI]                       MCG        4245     A
MCG                LX5    59-I.IIP                                               MCG        4246     A
MCG                PL     X5,SRM6     IF ^II[SI]    */ NOT ISSUED                MCG        4247     A
MCG                SA4    A5-B1       DI = LI - 1                                MCG        4248     A
MCG                LX6    B2,X4                                                  MCG        4249     A
MCG                PL     X6,SRM6     IF ^ST[DI]                                 MCG        4250     A
MCG                BX7    X1*X4                                                  MCG        4251     A
MCG                NZ     X7,SRM6     IF FP[DI] ! L2[DI] ! EQV[DI]               MCG        4252     A
MCG                SA2    A4-B1       R2 = DI - 1                                MCG        4253     A
MCG                AX7    B5,X2                                                  MCG        4254     A
MCG                SB4    X7                                                     MCG        4255     A
MCG                GE     B4,B1,SRM6  IF RF[R2] > 1                              MCG        4256     A
MCG                LX4    58-D.TYP                                               MCG        4257     A
MCG                MI     X4,SRM8     IF TYPE[DI] = III                          MCG        4258     A
MCG                                                                               MCG        4259     A
MCG       SRM6     LX6    B6,X3       UL = [UI]                                  MCG        4260     A
MCG                SA3    A3+B1       UI = UI + 1                                MCG        4261     A
MCG                PL     X6,SRM4     IF ^LAST[UL]                               MCG        4262     A
MCG                                                                               MCG        4263     A
MCG                RJ     OTS         OUTPUT A TEMP STORE                        MCG        4264     A
MCG                                                                               MCG        4265     A
MCG       SRM7     SA2    T0                                                     MCG        4266     A
MCG                ZR     X2,SRM7A    IF T0 = 0     */ NO INSTRUCTION TO ISSUE   MCG        4267     A
MCG                RJ     ISI         ISSUE SELECTED INSTRUCTION                 MCG        4268     A
MCG                SA7    UOR                                                    MCG        4269          I
          -FCC0584
MCG                                                                               MCG        4270     A
MCG       SRM7A    SA1    R0                                                     MCG        4271     A
MCG                RJ     GLI         GENERATE A *LD* OF SAVED RESULT            MCG        4272     A
MCG                EQ     SRM                                                    MCG        4273     A
MCG                                                                               MCG        4274     A
MCG       SRM8     BX6    X2                                                     MCG        4275     A
MCG                SA6    DRI+1       DRI(2) = [R2]                              MCG        4276     A
MCG                EQ     SRM7                                                   MCG        4277     A
MCG                                                                               MCG        4278     A
MCG       SRMA     BFMW   D,(FP,L2,EQV)                                          MCG        4279     A
MCG       GOT      TITLE  GOT - GET OT. ORDINAL                                  MCG        4280     A
MCG      **        GOT - GET OT. CELL ( TEMPORARY LOCATION ) FOR STORE TO TEMP   MCG        4281     A
MCG      *                                                                        MCG        4282     A
MCG      *         ENTRY  (X5) = RN , R-NUMBER TO BE SAVED IN THE OT.            MCG        4283     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG      *                                                                  MCG        4284    A
MCG      *         EXIT   (X6) = IHWORD ( R2 ) OF AN AVAILABLE LOCATION     MCG        4285    A
MCG                                                                         MCG        4286    A
MCG       GOT      ROUTINE                                                  MCG        4287    A
MCG                SA1    M.OT                                             MCG        4288    A
MCG                SA2    L.OTI                                            MCG        4289    A
MCG                IX6    X1-X2                                            MCG        4290    A
MCG                MI     X6,GOT1     IF M.OT < L.OTI                      MCG        4291    A
MCG                ALLOC  OTI,30B     ALLOC( OTI , 30B )                   MCG        4292    A
MCG                RJ     AIP         ADJUST *PIT* POINTERS                MCG        4293    A
MCG                SA1    M.OT                                             MCG        4294    A
MCG                                                                         MCG        4295    A
MCG       GOT1     SA2    O.OTI                                            MCG        4296    A
MCG                SB2    X1                                               MCG        4297    A
MCG                MX7    -I.USESL                                         MCG        4298    A
MCG                SA3    X2-1        OI = O.OTI - 1                       MCG        4299    A
MCG                SA7    X2+B2       [O.OTI+M.OT] = -0  */ SEARCH TERMINATOR MCG     4300    A
MCG                LX7    I.USESP                                          MCG        4301    A
MCG                                                                         MCG        4302    A
MCG      *         SEARCH FOR AN OT. CELL THAT IS NO LONGER BEING USED      MCG        4303    A
MCG                                                                         MCG        4304    A
MCG       GOT2     SA3    A3+B1       OI = OI + 1                          MCG        4305    A
MCG                MI     X3,GOT3     IF [OI] < 0   */ END OF TABLE        MCG        4306    A
MCG                SA2    "TB"+X3     LI = BASE + RI[OI]                   MCG        4307    A
MCG                BX1    -X7*X2                                           MCG        4308    A
MCG                NZ     X1,GOT2     IF USES[LI] " 0                      MCG        4309    A
MCG                                                                         MCG        4310    A
MCG                UX3    B2,X3       CA = UEXP(OI)                        MCG        4311    A
MCG                EQ     GOT4                                             MCG        4312    A
MCG                                                                         MCG        4313    A
MCG       GOT3     SX6    B2+B1       CA = M.OT                            MCG        4314    A
MCG                SA6    M.OT        M.OT = CA + 1                        MCG        4315    A
MCG                                                                         MCG        4316    A
MCG       GOT4     PX7    B2,X5                                            MCG        4317    A
MCG                SA4    =XOT.                                            MCG        4318         I
          -CCG8
MCG                SX6    B2                                               MCG        4319    A
MCG                SA7    A3          [OI] = PACK( CA , R-NUM )            MCG        4320    A
MCG       .PLI     IFNE   HC.ID,3     ^ PL/I                               CCG8        169    A
MCG                SA4    =XOT.                                            CCG8        170         I
          -CCG11
MCG                SA4    =XS$OT                                           CCG11       130    A
MCG                LX6    IH.CAP                                           MCG        4321    A
MCG                LX4    IH.IHP                                           MCG        4322    A
MCG                BX6    X4+X6       IHWORD = IHW(0,CA,IH)                MCG        4323    A
MCG       .PLI     ELSE                                                     CCG8        171    A
MCG                SA2    =XCO.PN                                          CCG8        172         I
          -CCGNC
MCG                SA2    =XHO$PN                                          CCGNC       791    A
MCG                SA3    =XO.ADT                                          CCG8        173         I
          -CCGNC
MCG                SA3    =XO$ADT                                          CCGNC       792    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                IX7    X2+X3                                          CCG8        174     A
MCG                SA3    X7                                             CCG8        175     A
MCG                SX4    B1                                             CCG8        176     A
MCG                SX6    B2+X3       CA = CA + LOW18[ADT(CO.PN)]        CCG8        177         I
          -CCGNC
MCG                SX6    B2+X3       CA = CA + LOW18[ADT(HO$PN)]        CCGNC       793     A
MCG                LX4    IH.RFP                                         CCG8        178     A
MCG                LX6    IH.CAP                                         CCG8        179     A
MCG                BX6    X4+X6       IHWORD = IHW(1,CA,0)               CCG8        180     A
MCG       .PLI     ENDIF                                                 CCG8        181     A
MCG                EQ     GOT                                            MCG        4324     A
MCG       SRM      SPACE  3                                              MCG        4325     A
MCG       OTS      TITLE  OTS - OUTPUT TEMP STORE                        MCG        4326     A
MCG      **        OTS - OUTPUT TEMP STORE                               MCG        4327     A
MCG      *                                                              MCG        4328     A
MCG      *         ENTRY  (B3) = REGNO OF REGISTER TO BE STORED OUT      MCG        4329     A
MCG      *                (R0) = TXT INDEX OF INSTRUCTION                MCG        4330     A
MCG      *                                                              MCG        4331     A
MCG      *         EXIT   (DRI+1) = IH WORD OF SAVED RESULT              MCG        4332     A
MCG                                                                     MCG        4333     A
MCG       OTS      ROUTINE                                               MCG        4334     A
MCG                SB2    OC.ST                                          MCG        4335     A
MCG                SA5    R0          R = [R0]                           MCG        4336     A
MCG                SA2    =XF.RDT+B2                                     MCG        4337     A
MCG                PX6    B2,X5                                          MCG        4338     A
MCG                SX4    B1                                             MCG        4339     A
MCG                SA6    DRI         DRI(0) = TYII(OC.ST,0,0,R)         MCG        4340     A
MCG                LX4    D.SRP                                          MCG        4341     A
MCG                BX7    X4+X2                                          MCG        4342     A
MCG                SA7    A6+2        DRI(2) = F.RDT(OC.ST) ! SR         MCG        4343     A
MCG                SX6    B3                                             MCG        4344     A
MCG                SA6    A7+B1       DRI(3) = REGNO                     MCG        4345     A
MCG                RJ     GOT         DRI(1) = GOT(R)                    MCG        4346     A
MCG                S"DI"  DRI+2       DI = DRI(2)                        MCG        4347     A
MCG                SA5    A4+B1                                          MCG        4348     A
MCG                SB3    X5                                             MCG        4349     A
MCG                SA6    A4-B1                                          MCG        4350     A
MCG                SB4    X5                                             MCG        4351     A
MCG                RJ     OEI         OUTPUT THE INSTRUCTION             MCG        4352     A
MCG                EQ     OTS                                            MCG        4353     A
MCG       GLI      SPACE  3                                              MCG        4354     A
MCG      **        GLI - GENERATE A *LD* OF A SAVED RESULT               MCG        4355     A
MCG      *                                                              MCG        4356     A
MCG      *         ENTRY  (X1) = R , *TXT* INDEX OF INSTRUCTION          MCG        4357     A
MCG      *                (DRI+1) = IH WORD OF SAVED RESULT              MCG        4358     A
MCG                                                                     MCG        4359     A
MCG       GLI      ROUTINE                                               MCG        4360     A
MCG                RJ     RIL         RESET R FOR REISSUE                MCG        4361     A
MCG                SB6    "TB"-B1                                        MCG        4362     A
MCG                SA3    DRI+1                                          MCG        4363     A
MCG                S"DI"  B6+B3       DI = BASE-1 + R                    MCG        4364     A
MCG                SB2    OC.LD                                          MCG        4365         I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
          -FCC0584
MCG                SX5    B3                                                    MCG        4366          I
          -FCC0584
MCG                SA2    =XIT.                                                 FCC0584      15     A
MCG                SB2    OC.LD                                                 FCC0584      16     A
MCG                BX7    X2-X3                                                 FCC0584      17     A
MCG                SX2    X7                                                    FCC0584      18     A
MCG                NZ     X2,GLI1     OC = IF( IH[DRI+1] " IT. ) THEN OC.LD     FCC0584      19     A
MCG                SB2    OC.TLD                    ELSE OC.TLD                 FCC0584      20     A
MCG       GLI      SX5    B3                                                    FCC0584      21     A
MCG                BX7    X3                                                    MCG        4367     A
MCG                SA1    F.RDT+B2                                              MCG        4368     A
MCG                SA2    GLIA                                                  MCG        4369     A
MCG                PX6    B2,X5                                                 MCG        4370     A
MCG                SA7    "DI"-B1     [DI-1] = [DRI+1]  */ IH WORD              MCG        4371     A
MCG                MX5    -D.STRSL                                              MCG        4372     A
MCG                SA6    A7-B1       [DI-2] = TYII(OC.LD,0,0,R)                MCG        4373          I
          -FCC0584
MCG                SA6    A7-B1       [DI-1] = TYII(OC,0,0,R)                   FCC0584      22     A
MCG                BX1    X1+X2                                                 MCG        4374     A
MCG                LX5    D.STRSP                                               MCG        4375     A
MCG                BX3    -X5*X4                                                MCG        4376     A
MCG                BX7    X1+X3       [DI] = RDT(OC.LD) ! ZP ! SR ! STRS[DI]    MCG        4377          I
          -FCC0584
MCG                BX7    X1+X3       [DI] = RDT(OC) ! ZP ! SR ! STRS[DI]       FCC0584      23     A
MCG                SA7    "DI"                                                  MCG        4378     A
MCG                LX4    59-D.PSP                                              MCG        4379     A
MCG                PL     X4,GLI      IF ^PS[DI]                                MCG        4380     A
MCG       GLI      SPACE  1,10                                                  MCG        4381     A
MCG      *         GENERATE AN *XMT* TO MOVE THE RESULT TO A STORE REGISTER     MCG        4382     A
MCG      *         THE STRATEGY IS TO MOVE THE ORIGINAL INSTRUCTION TO THE END  MCG        4383          I
          -FCC0881
MCG      *         OF *TXT* AND TO REPLACE THE ORIGINAL BY AN *XMT*.            MCG        4384          I
          -FCC0881
MCG                                                                             MCG        4385          I
          -FCC0881
MCG                SB2    "PI"                                                  MCG        4386          I
          -FCC0881
MCG                SA5    L.TXT       LT = L.TXT    */ SAVE PREVIOUS LENGTH     MCG        4387          I
          -FCC0881
MCG                MX3    -I.LINKL                                              MCG        4388          I
          -FCC0881
MCG                NZ     B2,GLI2           IF PI " 0                           MCG        4389          I
          -FCC0881
MCG                BX6    X5                                                    MCG        4390          I
          -FCC0881
MCG                SA6    ICL         ICL = LT                                  MCG        4391          I
          -FCC0881
MCG                EQ     GLI3                                                  MCG        4392          I
          -FCC0881
MCG       GLI2     SA1    "TB"+B2                                               MCG        4393          I
          -FCC0881

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                 BX2    X3*X1                                                    MCG        4394        I
          -FCC0881
MCG                 IX6    X2+X5       LINK[BASE+PI] = LT                           MCG        4395        I
          -FCC0881
MCG                 SA6    A1                                                       MCG        4396        I
          -FCC0881
MCG                                                                                 MCG        4397        I
          -FCC0881
MCG        GLI3     SX7    B3                                                       MCG        4398        I
          -FCC0881
MCG                 SA7    GLIB                                                     MCG        4399        I
          -FCC0881
MCG                 SA2    SIW                                                      MCG        4400        I
          -FCC0881
MCG                 NZ     X2,GLI4     IF SIW " 0                                   MCG        4401        I
          -FCC0881
MCG                 ALLOC  TREE,1                                                   MCG        4402        I
          -FCC0881
MCG                 SX6    X3-1        SIW = L.TREE - 1                             MCG        4403        I
          -FCC0881
MCG                 SA6    SIW                                                      MCG        4404        I
          -FCC0881
MCG        GLI4     ALLOC  TXT,4                                                    MCG        4405        I
          -FCC0881
MCG                 RJ     AIP         ADJUST INSTRUCTION POINTERS                  MCG        4406        I
          -FCC0881
MCG                                                                                 MCG        4407        I
          -FCC0881
MCG                 SA1    GLIB                                                     MCG        4408        I
          -FCC0881
MCG                 SB3    X1                                                       MCG        4409        I
          -FCC0881
MCG                 SB4    X5                                                       MCG        4410        I
          -FCC0881
MCG                 SA3    B3+X2       T0 = O.TXT + R                               MCG        4411        I
          -FCC0881
MCG                 BX4    X0*X3                                                    MCG        4412        I
          -FCC0881
MCG                 IX6    X4+X5       RI[T0] = LT                                  MCG        4413        I
          -FCC0881
MCG                 SA6    B4+X2       TI = O.TXT + LT;  [TI] = [T0]                MCG        4414        I
          -FCC0881
MCG                 SA3    A3+B1                                                    MCG        4415        I
          -FCC0881
MCG                 BX7    X3          [TI+1] = [T0+1]                              MCG        4416        I
          -FCC0881
MCG                 SA7    A6+B1                                                    MCG        4417        I
          -FCC0881
MCG                 MX3    -D.STRSL                                                 MCG        4418        I
          -FCC0881
MCG                 SA4    A3+B1       D0 = T0 + 2                                  MCG        4419        I
          -FCC0881

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                 LX3    D.STRSP                                                  MCG          4420          I
          -FCC0881
MCG                 BX6    -X3*X4       S = STRS[D0]                                MCG          4421          I
          -FCC0881
MCG                 BX7    X3*X4        STRS[D0] = 0                                MCG          4422          I
          -FCC0881
MCG                 SA7    A7+B1        [TI+2] = [D0]                               MCG          4423          I
          -FCC0881
MCG                 SA4    A4+B1        L = D0 + 1                                  MCG          4424          I
          -FCC0881
MCG                 SA3    SXBA                                                     MCG          4425          I
          -FCC0881
MCG                 SA2    SIW                                                      MCG          4426          I
          -FCC0881
MCG                 BX7    -X3*X4                                                   MCG          4427          I
          -FCC0881
MCG                 SB5    X2                                                       MCG          4428          I
          -FCC0881
MCG                 LX2    I.INDXP                                                  MCG          4429          I
          -FCC0881
MCG                 SX3    B1                                                       MCG          4430          I
          -FCC0881
MCG                 BX7    X2+X7        INDX[L] = SIW                               MCG          4431          I
          -FCC0881
MCG                 LX3    I.USESP                                                  MCG          4432          I
          -FCC0881
MCG                 BX7    X3+X7        USES[L] = 1                                 MCG          4433          I
          -FCC0881
MCG                 SA7    A7+B1        [TI+3] = [L]                                MCG          4434          I
          -FCC0881
MCG                 LX3    U.LASTP-I.USESP                                          MCG          4435          I
          -FCC0881
MCG                 SA2    O.TREE                                                   MCG          4436          I
          -FCC0881
MCG                 BX7    X1+X3                                                    MCG          4437          I
          -FCC0881
MCG                 SA7    X2+B5        [O.TREE+SIW] = SIT(1,0,0,R)                 MCG          4438          I
          -FCC0881
MCG                                                                                 MCG          4439          I
          -FCC0881
MCG       *         NOW SETUP AN *XMT* WHERE THE ORIGINAL INSTRUCTION WAS           MCG          4440          I
          -FCC0881
MCG                                                                                 MCG          4441          I
          -FCC0881
MCG                 SB2    OC.XMT                                                   MCG          4442          I
          -FCC0881
MCG                 SA2    F.RDT+B2                                                 MCG          4443          I
          -FCC0881
MCG                 BX7    X6+X2                                                    MCG          4444          I
          -FCC0881
MCG                 SA7    A4-B1        [T0+2] = RDT(OC.XMT) ! S                    MCG          4445          I
          -FCC0881

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                MX6    0                                                    MCG        4446          I
          -FCC0881
MCG                SA6    A7-B1       [T0+1] = 0                               MCG        4447          I
          -FCC0881
MCG                LX5    R1.RJP                                               MCG        4448          I
          -FCC0881
MCG                BX2    X5+X1                                                MCG        4449          I
          -FCC0881
MCG                PX7    B2,X2       [T0] = TYI(OC.XMT,LT,0,R)                MCG        4450          I
          -FCC0881
MCG                SA7    A6-B1                                                MCG        4451          I
          -FCC0881
MCG                SA3    GLIC                                                 MCG        4452          I
          -FCC0881
MCG                BX4    -X3*X4      IC,RI,LINK[L] = 0                        MCG        4453          I
          -FCC0881
MCG                SX5    B1                                                   MCG        4454          I
          -FCC0881
MCG                IX6    X4+X5       NPRED[L] = 1                             MCG        4455          I
          -FCC0881
MCG                SA6    A4                                                   MCG        4456          I
          -FCC0881
MCG                                                                            FCC0881     235     A
MCG                SA6    R0          R0 = R                                   FCC0881     236     A
MCG                RJ     SXB         SETUP A *XMT* BACK TO THE *ST-REG*       FCC0881     237     A
MCG                EQ     GLI                                                  MCG        4457     A
MCG                                                                            MCG        4458     A
MCG       GLIA     BFMW   D,(ZP,SR)                                            MCG        4459     A
MCG       GLIB     BSS    1                                                    MCG        4460          I
          -FCC0881
MCG       GLIC     BFMW   I,(IC,LINK)                                          MCG        4461          I
          -FCC0881
MCG       OEI      TITLE  OEI - OUTPUT EXTRA INSTRUCTION                       MCG        4462     A
MCG      **        OEI - OUTPUT EXTRA INSTRUCTION                              MCG        4463     A
MCG      *                                                                     MCG        4464     A
MCG      *         ENTRY  (A4,X4) = DI, [DI] OF DUMMY INSTRUCTION              MCG        4465     A
MCG      *                (B3) = REGNO OF RESULT REG OF INSTRUCTION            MCG        4466     A
MCG      *                (B4) = REGO , REGNO OF OPERAND REGISTER              MCG        4467     A
MCG      *                                                                     MCG        4468     A
MCG      *         CALLS  ASC, SII                                             MCG        4469     A
MCG                                                                            MCG        4470     A
MCG       OEI      ROUTINE                                                     MCG        4471     A
MCG                SA2    T                                                    MCG        4472     A
MCG                SA3    TRA+B4                                               MCG        4473     A
MCG      +         IX6    X2-X3                                                MCG        4474     A
MCG                PL     X6,*+1      T(IS) = MAX( T , TRA(REGO) )             MCG        4475     A
MCG                BX2    X3                                                   MCG        4476     A
MCG                LX4    -D.F1P                                               MCG        4477     A
MCG                SB6    "DI"+B1                                              MCG        4478     A
MCG                MX7    -D.F1L                                               MCG        4479     A
MCG                BX6    -X7*X4                                               MCG        4480     A
MCG                SB2    X6                                                   MCG        4481     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG              SX1    B6-"TB"                                                MCG        4482    A
MCG              MX5    -18                                                    MCG        4483    A
MCG              SB4    X2          T(EX) = T(IS)                              MCG        4484    A
MCG              BX6    -X5*X1                                                 MCG        4485    A
MCG              SB5    X2                                                     MCG        4486    A
MCG              PX7    B2,X6                                                  MCG        4487    A
MCG              SA7    PBI+1       FU[PBI(2)] = F1[DI]                        MCG        4488    A
MCG              LX4    D.F1P                                                  MCG        4489    A
MCG              PX6    B3,X2       REG[PBI] = REGNO                           MCG        4490    A
MCG              SA6    A7-B1                                                  MCG        4491    A
MCG              RJ     ASC         ADVANCE THE CLOCK                          MCG        4492    A
MCG              SA5    "DI"-2      R1 = DI - 2                                MCG        4493    A
MCG              RJ     SII         SAVE THE INSTRUCTION IN *PIT*              MCG        4494    A
MCG              EQ     OEI                                                    MCG        4495    A
MCG       AIS    SPACE  3           AIS                                        MCG        4496    A
MCG      **      AIS - ALLOCATE INSTRUCTION SPACE FOR EXTRA INSTRUCTIONS       MCG        4497    A
MCG                                                                            MCG        4498    A
MCG       AIS    ROUTINE                                                        MCG        4499    A
MCG              SA1    EIW                                                    MCG        4500    A
MCG              SX6    X1+B1       EIW = EIW + 1                              MCG        4501    A
MCG              SA6    A1                                                     MCG        4502    A
MCG              MI     X6,AIS      IF EIW < 0                                 MCG        4503    A
MCG              SA1    N.INST                                                 MCG        4504    A
MCG              AX1    2                                                      MCG        4505    A
MCG              BX6    -X1                                                    MCG        4506    A
MCG              SA6    A6          EIW = -N.INST/4                            MCG        4507    A
MCG              ALLOC  PIT,X1      ALLOC( PIT , N.INST/4 )                    MCG        4508    A
MCG              RJ     AIP         ADJUST POINTERS                            MCG        4509    A
MCG              EQ     AIS                                                    MCG        4510    A
MCG       AIP    SPACE  3,14        AIP                                        MCG        4511    A
MCG      **      AIP - ADJUST INSTRUCTION POINTERS ( AFTER A TABLE MANAGER CALL MCG        4512    A
MCG      *                                                                     MCG        4513    A
MCG      *       EXIT   (B7) = O.TXT + 3                                       MCG        4514    A
MCG      *                                                                     MCG        4515    A
MCG      *       USES   X - 1, 4, 6, 7 , A - 1, 4, 6, 7                        MCG        4516    A
MCG                                                                            MCG        4517    A
MCG       AIP0   SA1    O.TXT                                                  MCG        4518    A
MCG              SB7    X1+3        BASE = O.TXT + 3                           MCG        4519    A
MCG                                                                            MCG        4520    A
MCG       AIP    ROUTINE                                                        MCG        4521    A
MCG              SA1    O.PIT                                                  MCG        4522    A
MCG              SA4    IIP+1                                                  MCG        4523    A
MCG              IX6    X1-X4                                                  MCG        4524    A
MCG              ZR     X6,AIP0     IF IIP(2) = O.PIT  */ *PIT* NOT MOVED      MCG        4525    A
MCG              BX7    X1                                                     MCG        4526    A
MCG              SA1    A4-B1                                                  MCG        4527    A
MCG              IX6    X1+X6       IIP = IIP - IIP(2) + O.PIT  */ RESET POINTER MCG       4528    A
MCG              SA7    A4          IIP(2) = O.PIT    */ SAVE NEW BASE         MCG        4529    A
MCG              SA6    A1                                                     MCG        4530    A
MCG              EQ     AIP0                                                   MCG        4531    A
MCG                                                                            MCG        4532    A
MCG              QUAL                                                           MCG        4533    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

```
MCG                                                                    MCG        4534     A
MCG                END                                                 MCG        4535     A
```

SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - MCG

IDENTIFIER	TOTAL	ACTIVE
MCG	4535	4145
FCC0393	34	34
FCC0448	11	11
FCC0502	15	15
FCC0584	19	19
FCC0603	32	32
FCC0625	10	9
FCC0641	61	61
FCC0719	1	1
FCC0881	230	221
FEAT155A	20	18
FCC1056	39	39
FCC1126	6	6
FCC1442	9	9
FCC1451	13	10
RSM2763X	2	2
FCC1555X	86	83
FCC1580X	5	5
CCG1	5	1
CCG2	9	8
CCG3	45	44
CCG4	2	0
MCGF1	11	9
CCG7	7	4
CCG8	27	20
FCC1655X	23	23
FCC1656X	36	36
CCGNC	24	24
CCG11	7	6
FCC1845X	1	1
CCG16	11	8
CCG18	2	2
CCG19	153	145
CCGA006	30	30
FCC2038	1	1
CCGA019	135	128
CCGA023	164	161
CCGA034	2	2
FCCA546X	4	4
CCGA066	1	1
CCGA073	7	7
CCG0008	32	32
CCG0086	2	2
CCG0110	69	69
CCG0157	2	2

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN MCG

                CCGA093                    13        13
                CCG0191                    26        26
                CCG0196                     2         2

                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

BDT             *DECK     BDT                                                BDT            1     A
BDT                       IDENT  BDT                                         BDT            2     A
BDT              BDT      TITLE  BDT - FORM DEPENDENCY GRAPH                 BDT            3     A
BDT             *CALL     SSTCALL                                            BDT            4     A
BDT              B=BDT    RPVDEF                                             BDT            5     A
BDT              SNAP..   OPSYN  SNAP                                        CCGA017        8     A
BDT                       PURGMAC  SNAP                                      CCGA017        9     A
BDT                       MACRO  SNAP,NAME,FWA,LEN,LWA,AREG,LL,UL,INC,NUSE,UF CCGA017      10     A
BDT              NAME     SNAP.. FWA,LEN,LWA,NOREG,LL,UL,INC,NUSE,UF         CCGA017       11     A
BDT              SNAP     ENDM                                               CCGA017       12     A
BDT                                                                          CCGA017       13     A
BDT                       PASS2TM                                            BDT            6          I
                -CCG1
BDT                       TABLES TXT,TREE                                    BDT            7          I
                -CCG1
BDT                                                                          CCG1         286     A
BDT                       EXT    O.TREE,L.TREE,O.TXT,L.TXT                   CCG1         287          I
                -CCGNC
BDT                       TABLES (IIT,TET,TREE,TXT)                          CCGNC        794     A
BDT             *IF       DEF,BDT,1                                          BDT            8     A
BDT                       TRACER BDT                                         BDT            9     A
BDT              BDT      SPACE  3                                           BDT           10     A
BDT             **        BDT - BUILD DEPENDENCY TREE ( FORM DEPENDENCY GRAPH ) BDT         11     A
BDT             *         BDT FORMS THE DEPENDENCY GRAPH OF A SEQUENCE OF *IL*  BDT         12     A
BDT             *         INSTRUCTIONS WHICH TAKES INTO ACCOUNT THE OPERAND, DATA BDT       13     A
BDT             *         INTERFERENCE, REGISTER REDEFINITION AND BLOCK CONSTRAINT BDT      14     A
BDT             *         DEPENDENCIES ( LINKS ) OF THE INSTRUCTIONS.  IT ALSO COMPUTES BDT 15     A
BDT             *         THE PERT LATE START TIMES FOR EACH INSTRUCTION, AND SORTS THE BDT 16     A
BDT             *         THE SUCCESSOR INDEX LISTS ACCORDING TO THEM.          BDT        17     A
BDT             *                                                               BDT        18     A
BDT             *         S.I. JASIK - FEB 1973                                 BDT        19     A
BDT             *                                                               BDT        20     A
BDT             *         ENTRY  (X6) = TREE MODE = 0/1 - PRIORITY CALC / JAM MODE BDT     21     A
BDT             *                (SIOCALL) = 1 IF CALL FROM *RIO* FOR A PARTIAL TREE BDT 22     A
BDT             *                *TXT* - INSTRUCTION SEQUENCE, LINK WORDS ARE ZERO. BDT 23     A
BDT             *                                                               BDT        24     A
BDT             *         EXIT   *TREE* = SUCCESSOR INDEX TABLE IN "U." FORMAT  BDT        25     A
BDT             *                LINK WORDS OF INSTRUCTIONS IN "I." FORMAT      BDT        26     A
BDT             *                N.INST, N.ST , SEQTIME SET                     BDT        27     A
BDT                       SPACE  3                                               BDT        28     A
BDT                       USE    /MCG/                                           BDT        29     A
BDT              N.INST   BSS    1          NUMBER OF INSTRUCTIONS ( L.TXT/4 )  BDT        30     A
BDT              N.ST     BSS    1          NUMBER OF STORES IN *TXT*           BDT        31     A
BDT              SEQTIME  BSS    1          CRITICAL PATH TIME ( CALCULATED IN *CIP* ) BDT 32     A
BDT                       USE    0                                               BDT        33     A
BDT              DEBUG    SPACE  3                                               BDT        34     A

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT      **        DEBUGGING FACILITIES                                          BDT          35     A
BDT      *                                                                       BDT          36     A
BDT      *         BASIC SNAPS - *IL* SEQUENCE AND *SIT* ON EXIT                 BDT          37     A
BDT      *         TRACER BDT                                                    BDT          38     A
BDT      *                                                                       BDT          39     A
BDT      *         TO SNAP THE FINAL DEPENDENCY TREE                             BDT          40     A
BDT      *         TRACER TREE                                                   BDT          41     A
BDT      *                                                                       BDT          42     A
BDT      *         TO TRACE EACH PHASE -                                         BDT          43     A
BDT      *         TRACER (RNI,IH,FIL,FOL,FTL,FJL)                               BDT          44     A
BDT                                                                              BDT          45     A
BDT      **        TREE - SNAPSHOT DUMP OF THE DEPENDENCY TREE                   BDT          46     A
BDT      *         NOTE - TREE MACRO MAY BE CALLED BETWEEN FIL2 AND FTL5 ONLY    BDT          47     A
BDT                                                                              BDT          48     A
BDT       TREE     MACRO  LAB                                                    BDT          49     A
BDT                IF     DEF,/DEBUG/LAB,1                                       BDT          50     A
BDT                SNAP   -A7,-TREEL,,1,200,1,TREE-LAB                           BDT          51          I
         -FEAT155F
BDT       TREE-LAB SNAP   *A7,*TREEL,,,1,200,1                                   FEAT155F    261     A
BDT                ENDM                                                          BDT          52     A
BDT       TABLES   TITLE  TABLE FORMATS                                          BDT          53     A
BDT                LIST   -R                                                     BDT          54     A
BDT                                                                              BDT          55     A
BDT      **        DEPENDENCY TREE ( FIL,FOL,FTL, ETC)                           BDT          56     A
BDT                                                                              BDT          57     A
BDT                DESCRIBE T.,60     TREE(SUCC,PRED)                            BDT          58          I
         -FCC0448
BDT                DESCRIBE T.,60     TREE(EQVL,SUCC,PRED)                       FCC0448     158     A
BDT                DEFINE 18                                                     BDT          59     A
BDT       SUCC     DEFINE 18          INDEX TO *TXT* OF SUCCESSOR                BDT          60     A
BDT                DEFINE 1                                                      BDT          61     A
BDT       EQVL     DEFINE 1           DATA INTERFERENCE LINK                     BDT          62          I
         -FCC0448
BDT                DEFINE 4                                                      BDT          63          I
         -FCC0448
BDT       RDL      DEFINE 1           REGISTER REDEFINITION LINK                 FCC0448     159     A
BDT       EQVL     DEFINE 1           DATA INTERFERENCE LINK                     FCC0448     160     A
BDT                DEFINE 3                                                      FCC0448     161     A
BDT       PRED     DEFINE 18          INDEX TO *TXT* OF PREDECESSOR              BDT          64     A
BDT       IHINFO   SPACE  3                                                       BDT          65     A
BDT      **        IH INFO TABLE ( FIL )                                         BDT          66     A
BDT                                                                              BDT          67     A
BDT                DESCRIBE IH.,60    IHINFO(BITS,R2W)                           BDT          68     A
BDT       LD       DEFINE 1                                                      BDT          69     A
BDT       ST       DEFINE 1                                                      BDT          70     A
BDT       BM       DEFINE 1                                                      BDT          71     A
BDT                DEFINE 57          R2 WORD OF MEM REF INSTRUCTION             BDT          72     A
BDT       USESI    SPACE  3                                                       BDT          73     A
BDT      **        USES INDEX TABLE ( FINAL FORMAT OF TREE )                     BDT          74     A
BDT      *               SORTED ON *PRED* , *LST* AND *SUCC* FIELDS.             BDT          75     A
BDT                                                                              BDT          76     A
BDT                DESCRIBE U.,60     SIT(LAST,PRED,LST,SUCC)                    BDT          77     A

         0        1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT       LAST     DEFINE 1          "0 IF LAST EDGE OF A *PRED* GROUP                BDT          78     A
BDT       EQV      DEFINE 1          DATA INTERFERENCE LINK                           BDT          79          I
         -FCC0448
BDT                DEFINE 4                                                           BDT          80          I
         -FCC0448
BDT       RDL      DEFINE 1          REGISTER REDEFINITION LINK                       FCC0448     162     A
BDT       EQV      DEFINE 1          DATA INTERFERENCE LINK                           FCC0448     163     A
BDT                DEFINE 3                                                           FCC0448     164     A
BDT       PRED     DEFINE 18         INDEX TO *TXT* OF PREDECESSOR                    BDT          81     A
BDT       LST      DEFINE 18         LATE START TIME                                  BDT          82     A
BDT       SUCC     DEFINE 18         INDEX TO *TXT* OF SUCCESSOR                      BDT          83     A
BDT       INFO     SPACE  3                                                           BDT          84     A
BDT      **        I. - LINK WORD FORMAT FOR INSTRUCTION SCHEDULING ( BDT EXIT )      BDT          85     A
BDT                                                                                   BDT          86     A
BDT                DESCRIBE I.,60    UINFO(USES,INDX,PRI,NPRED)                       BDT          87     A
BDT                DEFINE 3                                                           BDT          88     A
BDT       USES     DEFINE 9          USES COUNT                                       BDT          89     A
BDT       INDX     DEFINE 15         ORDINAL TO SUCCESSOR LIST IN USES INDEX TBL      BDT          90     A
BDT       LST      DEFINE 15         LATE START TIME                                  BDT          91     A
BDT       NPRED    DEFINE 18         NUMBER OF UNISSUED PREDECESSORS                  BDT          92     A
BDT                                                                                   BDT          93     A
BDT       PRIOR    DEQU   LST        PRIORITY ( NEGATIVE OF LATE START TIME )         BDT          94     A
BDT       INIT     TITLE  INITIALIZATION                                              BDT          95     A
BDT      **        LOCAL STORAGE                                                      BDT          96     A
BDT                                                                                   BDT          97     A
BDT                USE    /TABLES/                                                    BDT          98          I
         -CCG1
BDT                USE    /CCG.SCR/                                                   CCG1        288          I
         -CCG18
BDT                USE    /CCGSCR/                                                    CCG18       175     A
BDT       BDTA     BSS    1          TREE BUILD MODE                                  BDT          99     A
BDT       EQVL     BSS    1          "0 IF JAM MODE AND EQUIV LINKS                   BDT         100     A
BDT       FBM      BSS    1          ORD OF FIRST BOUNDARY MARKER IN *TXT*            BDT         101     A
BDT       LBM      BSS    1          DESCR ADDRESS OF LAST *BM*                       BDT         102     A
BDT       FTREE    BSS    1          FWA OF TEMP TREE                                 BDT         103     A
BDT       TF       BSS    1          LWA+1 OF THE OPERAND LINKS ( TEMP TREE )         BDT         104     A
BDT       TREEL    BSS    1          LWA+1 OF THE TREE                                BDT         105     A
BDT       TSTC     BSS    1          *TST* CHAIN ( NBIP " 0 & SIOCALL = 2 ONLY )      FCC0514     301     A
BDT       REGS     BSS    24         REGISTER STORE ADDRESS TABLE                     BDT         106     A
BDT                USE    0                                                           BDT         107     A
BDT                                                                                   BDT         108     A
BDT                                                                                   BDT         109     A
BDT       SIOCALL  BSSZ   1          "0 IF CALL FROM *RIO*                            BDT         110     A
BDT                                                                                   BDT         111     A
BDT                LIST   R                                                           BDT         112     A
BDT                                                                                   FCC0514     302     A
BDT       PI.CNP   EQU    36         IP. CLASS NUMBER                                 FCC0514     303     A
BDT       D.RDP    EQU    D.MUCP     REDEF ST (FIL10,SIO3)                            FCC1126      13     A
BDT                                                                                   BDT         113     A
BDT       OC       MICRO  1,,/B3/                                                     BDT         114     A
BDT       BDT      SPACE  3                                                           BDT         115     A
BDT       BDT      ENTRY.                                                             BDT         116          I

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

                -CCGNC
      BDT        BDT      ENTRY. **,#                                                  CCGNC        795     A
      BDT                 QUAL   BDT                                                  BDT          117     A
      BDT                 SA1    L.TXT                                                BDT          118     A
      BDT                 SA6    BDTA        SAVE TREE BUILD MODE                     BDT          119     A
      BDT                 BX7    X1                                                   BDT          120     A
      BDT                 AX7    2                                                    BDT          121     A
      BDT                 SA7    N.INST      N.INST = L.TXT/4                         BDT          122     A
      BDT                 SX1    X1+100      L.TREE = L.TREE + 100  */ FUDGE FACTOR   CCG0188        6     A
      BDT                 ZR     X6,BDT2     IF BDTA = 0   */ NOT JAM MODE            BDT          123     A
      BDT                 SX1    X1+200B     L.TREE = L.TREE + 200B  */ FUDGE FACTOR  BDT          124     A
      BDT                                                                             BDT          125     A
      BDT        BDT2     ALLOC  TREE,X1     ALLOC( TREE , L.TREE )                   BDT          126     A
      BDT                 SX3    X3-1                                                 BDT          127     A
      BDT                 SA5    O.TXT                                                BDT          128     A
      BDT                 IX6    X2+X3                                                BDT          129     A
      BDT                 SA6    TREEL       TREEL = LWA+1 OF THE TEMP TREE           BDT          130     A
      BDT                                                                             BDT          131     A
      BDT                 SA0    X5          (A0) = O.TXT                             BDT          132     A
      BDT                 SB7    X2          (B7) = O.TREE                            BDT          133     A
      BDT                 SB2    B1+B1       (B2) = 2                                 BDT          134     A
      BDT                 SB4    4           (B4) = 4                                 BDT          135     A
      BDT                 SX7    0                                                    FCC0514      304         I
                -FCC1739X
      BDT                 SA7    TSTC        TSTC = 0                                 FCC0514      305         I
                -FCC1739X
      BDT                                                                             BDT          136     A
      BDT       *         THE TEMPORARY TREE IS FORMED IN *TREE* STARTING AT THE END  BDT          137     A
      BDT       *         OF THE TABLE AND WORKING TOWARDS THE BEGINNING.             BDT          138     A
      BDT       *         (A7) HOLDS THE ADDRESS OF THE LAST EDGE ADDED TO IT FROM    BDT          139     A
      BDT       *         *FIL3* TO *FTL7*.  DURING FIL THE *IHINFO* TABLE IS SETUP AT BDT         140     A
      BDT       *         THE BEGINNING OF *TREE*.                                    BDT          141     A
      BDT        FIL      TITLE  FIL - FORM DATA INTERFERENCE LINKS                   BDT          142     A
      BDT       **        FORM THE *IH* INFO TABLE                                    BDT          143     A
      BDT       *               60/*TXT* INDEX OF INSTRUCTION                         BDT          144     A
      BDT       *               60/*IH* INFO WORD                                     BDT          145     A
      BDT                                                                             BDT          146     A
      BDT        FIL      SA2    O.TREE                                               BDT          147     A
      BDT                 MX6    60                                                   BDT          148     A
      BDT                 SA6    X2          IH = O.TREE,  [IH] = -0                  BDT          149     A
      BDT                 MX0    -1                                                   BDT          150     A
      BDT                 MX1    -D.LDSTL                                             BDT          151     A
      BDT                 SB5    A0+B2       BASE = O.TXT + 2                         BDT          152     A
      BDT                 LX1    D.LDSTP                                              BDT          153     A
      BDT                 SB6    59-D.BMP                                             BDT          154     A
      BDT                 SB7    B0          N.ST = 0                                 BDT          155     A
      BDT                 SA4    A0+B2       DI = O.TXT + 2                           BDT          156     A
      BDT                 SA2    TREEL                                                BDT          157     A
      BDT                 SA3    BDTA                                                 BDT          158     A
      BDT                 MX7    0                                                    BDT          159     A
      BDT                 SA7    TSTC        TSTC = 0                                 FCC1739X       5     A
      BDT                 SA7    X2          T = TREEL;  [T] = 0                      BDT          160     A

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                 SX6    A0                                                    BDT          161     A
BDT                 MX5    -IH.BMP                                               BDT          162     A
BDT                 BX7    X5+X6                                                 BDT          163     A
BDT                 SA7    A7-B1       T = T - 1;  [T] = BM[IH] ! O.TXT          BDT          164     A
BDT                                                                              BDT          165     A
BDT       *         SET FLAG FOR FULL EQUIV LINKING ( BETWEEN CONDITIONAL JUMPS ) BDT          166     A
BDT       *         IF JAM MODE. ASSUME THAT CONDITIONAL JUMPS ARE TYPE III.      BDT          167         I
          -FCC1695X
BDT       *         IF JAM MODE OR CALLED FROM *RIO*.                             FCC1695X       5     A
BDT                                                                              BDT          168     A
BDT                 SX2    B1                                                    BDT          169     A
BDT                 LX2    D.BMP                                                 BDT          170         I
          -FCC1695X
BDT                 ZR     X3,FIL2     FEL = IF( BDTA = 0 ) THEN BM; ELSE *TY*   BDT          171         I
          -FCC1695X
BDT                 LX2    D.TYP-D.BMP                                           BDT          172         I
          -FCC1695X
BDT                 LX2    D.TYP       FET = TY      */TY IV BM"S ONLY           FCC1695X       6     A
BDT                 NZ     X3,FIL2     IF BDTA " 0                               FCC1695X       7     A
BDT                 SA3    SIOCALL                                               FCC1695X       8     A
BDT                 NZ     X3,FIL2     IF SIOCALL " 0                            FCC1695X       9     A
BDT                 LX2    D.BMP-D.TYP FEL = BM      */ DONT LINK ACROSS TY IIIJPS FCC1695X      10     A
BDT                 EQ     FIL2                                                  BDT          173     A
BDT                                                                              BDT          174     A
BDT       *         SCAN THE SEQUENCE AND FORM AN *IH* INFO TABLE ENTRY FOR EACH BDT          175     A
BDT       *         LD, ST AND BOUNDARY MARKER.                                   BDT          176     A
BDT                                                                              BDT          177     A
BDT        FIL1     SA5    A4-B1       R2 = DI - 1                               BDT          178     A
BDT                 LX4    -D.STP                                                BDT          179     A
BDT                 BX3    -X0*X4                                                BDT          180     A
BDT                 IFNE   .DAL,0,2                                 LEVEL 2      BDT          181     A
BDT                 LX4    D.STP-D.L2P                                           BDT          182     A
BDT                 BX3    -X4*X3                                                BDT          183     A
BDT                                                                              BDT          184     A
BDT                 SA6    A6+B1       IH = IH + 1;  [IH] = DI - BASE            BDT          185     A
BDT                 LX7    IH.STP-D.LDSTP                                        BDT          186     A
BDT                 SB7    B7+X3       N.ST = N.ST + (ST[DI]&^L2[DI])            BDT          187     A
BDT                 BX6    X7+X5                                                 BDT          188     A
BDT                 SA6    A6+B1       IH = IH + 1;  [IH] = IHINFO(LDST[DI],[R2]) BDT         189     A
BDT                 ZR     X3,FIL2     IF (ST[DI]&^L2[DI]) = 0                   BDT          190     A
BDT                 SX7    A4-B2                                                 BDT          191     A
BDT                 SA7    A7-B1       T = T - 1;  [T] = R1                      BDT          192     A
BDT                                                                              BDT          193     A
BDT        FIL2     SA4    A4+B4       DI = DI + 4                               BDT          194     A
BDT                 BX7    -X1*X4                                                BDT          195     A
BDT                 SX6    A4-B5                                                 BDT          196     A
BDT                 LX3    B6,X4                                                 BDT          197     A
BDT                 NZ     X7,FIL1     IF LDST[DI]                               BDT          198     A
BDT                 PL     X3,FIL2     IF ^BM[DI]                                BDT          199     A
BDT                 SA5    A4-B2       R1 = DI - 2                               BDT          200     A
BDT                 BX3    X2*X4                                                 BDT          201     A
BDT                 ZR     X3,FIL2     IF FEL[DI] = 0  */ JAM MODE & TY III BM   BDT          202     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                SX7    A4-B2                                                    BDT          203     A
BDT                SA6    A6+B1       IH = IH + 1;  [IH] = DI - BASE               BDT          204     A
BDT                UX3    "OC",X5                                                  BDT          205     A
BDT                MX6    -IH.BMP                                                  BDT          206     A
BDT                BX7    X6+X7                                                    BDT          207     A
BDT                SA6    A6+B1       IH = IH + 1;  [IH] = IHINFO(BM,0)            BDT          208     A
BDT                SA7    A7-B1       T = T - 1;  [T] = BM[IH] ! R1                BDT          209     A
BDT                NZ     "OC",FIL2   IF OC[R1] " 0                                BDT          210     A
BDT                                                                                BDT          211     A
BDT                SB6    A6          IHS = IH                                     BDT          212     A
BDT                SX6    B7                                                       BDT          213     A
BDT                SA6    N.ST                                                     BDT          214     A
BDT                                                                                BDT          215     A
BDT                IF     DEF,/DEBUG/IH,1                                          BDT          216     A
BDT                SNAP   -O.TREE,-B6,,1,100,1,IH-INFO                             BDT          217         I
          -FEAT155F
BDT       IH-INFO  SNAP   *O.TREE,B6                                               FEAT155F     262     A
BDT       CSP      SPACE  3,14                                                     BDT          218     A
BDT      **        CSP - COUNT NUMBER OF STORE PREDECESSORS PER BASIC BLOCK        BDT          219     A
BDT      *         AND SAVE THE COUNTS IN THE USES FIELD OF THE DESCR OF THE       BDT          220     A
BDT      *         BOS/UJP/RJX/LAB STARTING THE BLOCK.                             BDT          221     A
BDT                                                                                BDT          222     A
BDT                ZR     B7,CSP6     IF N.ST = 0   */ IF NO STORE PREDECESSORS    BDT          223     A
BDT                MX0    -R1.RIL                                                  BDT          224     A
BDT                SB5    B0          NSP = 0       */ NUMBER OF STORE PRED        BDT          225     A
BDT                SA5    A7+B1       TI = T + 1                                   BDT          226     A
BDT                BX1    -X0         RMIN = 177777B                               BDT          227     A
BDT                SB3    60-D.TYL                                                 BDT          228     A
BDT                SB7    T.SUCCP                                                  BDT          229     A
BDT                EQ     CSP2                                                     BDT          230     A
BDT                                                                                BDT          231     A
BDT       CSP1     SA5    A5+B1       TI = TI + 1                                  BDT          232     A
BDT                BX1    X3          RMIN = R                                     BDT          233     A
BDT                SB5    B5+1        NSP = NSP + 1                                BDT          234     A
BDT                                                                                BDT          235     A
BDT       CSP2     MI     X5,CSP4     IF BM[TI]                                    BDT          236     A
BDT                SA4    X5          R1 = [TI]                                    BDT          237     A
BDT                BX3    -X0*X4      R = RI[R1]                                   BDT          238     A
BDT                LX6    B7,X3                                                    BDT          239     A
BDT                BX7    X6+X5       [TI] = TREE(R,R1)                            BDT          240     A
BDT                IX4    X3-X1                                                    BDT          241     A
BDT                SA7    A5                                                       BDT          242     A
BDT                MI     X4,CSP1     IF R < RMIN                                  BDT          243     A
BDT                                                                                BDT          244     A
BDT      *         POSSIBLE PS REVERSAL, SCAN BACKWARDS TO FIND *RI* IN STORE.     BDT          245     A
BDT                                                                                BDT          246     A
BDT                SA4    A5-B1       TJ = TI - 1                                  BDT          247     A
BDT                SA5    A5+B1       TI = TI + 1                                  BDT          248     A
BDT       CSP3     AX6    B7,X4                                                    BDT          249     A
BDT                IX7    X6-X3                                                    BDT          250     A
BDT                ZR     X7,CSP2     IF SUCC[TJ] = R                              BDT          251     A
BDT                SA4    A4-B1       TJ = TJ - 1                                  BDT          252     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                PL     X4,CSP3     IF ^BM[TJ]                                    BDT          253     A
BDT                SB5    B5+B1       NSP = NSP + 1                                 BDT          254     A
BDT                EQ     CSP2                                                      BDT          255     A
BDT                                                                                BDT          256     A
BDT       *        BOUNDARY MARKER ENCOUNTERED, SAVE *NSP* IN IT IF TYPE = IV      BDT          257     A
BDT                                                                                BDT          258     A
BDT        CSP4    SA4    X5+B2       DI = [TI] + 2                                 BDT          259     A
BDT                SA5    A5+1        TI = TI + 1                                   BDT          260     A
BDT                LX4    58-D.TYP                                                  BDT          261     A
BDT                AX6    B3,X4                                                     BDT          262     A
BDT                ZR     X6,CSP5     IF TYPE[DI] = IV                              BDT          263     A
BDT                SX7    A4-B2                                                     BDT          264     A
BDT                SA7    A5-B1       [TI-1] = DI - 2 */ REMOVE BM BITS             BDT          265     A
BDT                EQ     CSP2                                                      BDT          266     A
BDT                                                                                BDT          267     A
BDT        CSP5    SX3    B5          USES[DI] = NSP                                BDT          268     A
BDT                MX7    -D.USESL                                                  BDT          269     A
BDT                LX4    2+D.TYP                                                   BDT          270     A
BDT                SB5    B0          NSP = 0                                       BDT          271     A
BDT                BX4    X7*X4                                                     BDT          272     A
BDT                BX6    X3+X4                                                     BDT          273     A
BDT                SA6    A4                                                        BDT          274     A
BDT                NZ     X5,CSP2     IF [TI] " 0                                   BDT          275     A
BDT                                                                                BDT          276     A
BDT        CSP6    SA1    TREEL                     */ REINTIALIZE THE TREE        BDT          277     A
BDT                SX7    0                                                         BDT          278     A
BDT                SA7    EQVL        EQVL = 0                                      BDT          279     A
BDT                SA7    X1          T = TREEL;  [T] = 0                           BDT          280     A
BDT                                                                                FCC0977       39     A
BDT       *        IF IN A LOOP, THEN CHECK *IHINFO* TABLE FOR *SRF* MEMORY        FCC0977       40     A
BDT       *        REFERENCES, AND INVENT AN *RF* FOR THEM.                        FCC0977       41     A
BDT       *        *SRF* MEMORY REFS ARE THE RESULT OF ADDRESS DIFFERENCING        FCC0977       42     A
BDT       *        SIMILAR *IP"S*, AND REFERENCES TO THE NON-BASE MEMBERS OF       FCC0977       43     A
BDT       *        THE CLASS DO NOT HAVE A REAL *RF* ASSOCIATED WITH THEM.         FCC0977       44     A
BDT                                                                                FCC0977       45     A
BDT                                                                                FCC0977       46     A
BDT                SA1    =XGRASRF                                                FCC0977       47           I
          -CCGNC
BDT                SA1    =XGRASRF#                                               CCGNC        796     A
BDT                ZR     X1,IRF5     IF GRASRF = 0 */ ^(IN LOOP & ADDR DIFF)       FCC0977       48     A
BDT                                                                                FCC0977       49     A
BDT                MX0    -IH.RFL                                                 FCC0977       50     A
BDT                SA1    SIOCALL                                                 FCC0977       51     A
BDT                LX0    IH.RFP                                                  FCC0977       52     A
BDT                MX7    -2*R1.RIL                                               FCC0977       53     A
BDT                SA5    B6          IH = IHS                                    FCC0977       54     A
BDT                LX7    R1.RIL                                                  FCC0977       55     A
BDT                SB3    A0                                                      FCC0977       56     A
BDT                SB5    59-IH.SRFP                                              FCC0977       57     A
BDT                SB7    B1          IRN = 1       */ INVENTED R-NUMBER INDEX    FCC0977       58     A
BDT                                                                                FCC0977       59     A
BDT       *        SEARCH *IHINFO* FOR *SRF* MEM REFS, AND INVENT A *RF*           FCC0977       60     A

          0        1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT       *          FOR EACH UNIQUE ORDERED PAIR - *(RJ,RK)* .                    FCC0977     61    A
BDT                                                                                FCC0977     62    A
BDT        IRF1      SA5     A5-B2       IH = IH - 2                               FCC0977     63    A
BDT                  LX6     B5,X5                                                 FCC0977     64    A
BDT                  PL      X6,IRF1     IF ^SRF[IH]                               FCC0977     65    A
BDT                  ZR      X5,IRF5     IF [IH] = 0   */ END OF *IHINFO*          FCC0977     66    A
BDT                                                                                FCC0977     67    A
BDT                  SA4     A5-B1       R = [IH-1]                                FCC0977     68    A
BDT                  SA2     B3+X4       R1W = TXT(R)                              FCC0977     69    A
BDT                  BX6     -X7*X2      KEY = RJRK[R1W]                           FCC0977     70    A
BDT                  SA6     A7-B7       T(-IRN) = KEY */ SET SEARCH TERMINATOR    FCC0977     71    A
BDT                  BX5     X0*X5                                                 FCC0977     72    A
BDT                  SA3     A7          TJ = T                                    FCC0977     73    A
BDT        IRF2      SA3     A3-B1       TJ = TJ - 1   */ SEARCH ORDERED PAIR LIST FCC0977     74    A
BDT                  BX2     X3-X6                                                 FCC0977     75    A
BDT                  NZ      X2,IRF2     IF [TJ] " KEY                             FCC0977     76    A
BDT                                                                                FCC0977     77    A
BDT                  SB4     A3                                                    FCC0977     78    A
BDT                  SB4     A7-B4       J = T - TJ                                FCC0977     79    A
BDT                  LT      B4,B7,IRF3  IF J < IRN                                FCC0977     80    A
BDT                  SB7     B7+B1       IRN = IRN + 1                             FCC0977     81    A
BDT        IRF3      SX2     B4                                                    FCC0977     82    A
BDT                  LX2     2                                                     FCC0977     83    A
BDT                  SX2     X2+B2                                                 FCC0977     84    A
BDT                  LX2     IH.RFP      JR = 4*J+2    */ INVENTED R-NUM           FCC0977     85    A
BDT                  BX6     X2+X5       RF[IH] = JR   */ SET IN *IHINFO*          FCC0977     86    A
BDT                  SA6     A5                                                    FCC0977     87    A
BDT                  SA4     A2+B1       R2 = TXT(R+1)                             FCC0977     88    A
BDT                  BX6     X0*X4       RF[R2] = 0                                FCC0977     89    A
BDT                  SA6     A4                                                    FCC0977     90    A
BDT                  NZ      X1,IRF1     IF SIOCALL " 0  */ CALL FROM *RIO*        FCC0977     91    A
BDT                                                                                FCC0977     92    A
BDT                  BX6     X2+X6       RF[R2] = JR                               FCC0977     93    A
BDT                  SA6     A6                                                    FCC0977     94    A
BDT                  EQ      IRF1                                                  FCC0977     95    A
BDT                                                                                FCC0977     96    A
BDT        IRF5      BSS     0                                                     FCC0977     97    A
BDT                  TRACE   IRF,RLIST                                             FCC0977     98    A
BDT                  IF      DEF,/DEBUG/IRF,1                                      FCC0977     99    A
BDT        IH-IRFX   SNAP    *O.TREE,*B6                                           FCC0977    100    A
BDT        FIL       EJECT                                                         BDT        281    A
BDT       **         SCAN THE *IH* INFO TABLE FOR INTERFERING LD/ST, ST/ST         BDT        282    A
BDT       *          COMBINATIONS AND FORM TREE LINKS FOR THEM.                    BDT        283    A
BDT                                                                                BDT        284    A
BDT                  MX0     -IH.IHL                                               BDT        285    A
BDT                  SA5     B6          IH = IHS                                  BDT        286    A
BDT                  SB5     59-IH.BMP                                             BDT        287    A
BDT                  SA1     BDTA                                                  BDT        288    A
BDT                  LX1     T.EQVLP     EQVB = SHIFT(BDTA,T.EQVLP)                BDT        289    A
BDT                                                                                BDT        290    A
BDT        FIL3      SA5     A5-B2       IH = IH - 2                               BDT        291    A
BDT                  LX6     B5,X5                                                 BDT        292    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT              ZR     X5,FIL6     IF [IH] = 0   */ IF TABLE TERMINATOR          BDT   293   A
BDT              MI     X6,FIL3     IF BM[IH]                                     BDT   294   A
BDT              SA4    A5          J = IH                                        BDT   295   A
BDT              SA3    A5-B1                                                     BDT   296   A
BDT              LX3    T.SUCCP                                                   BDT   297   A
BDT              IX2    X1+X3                                                     BDT   298   A
BDT                                                                               BDT   299   A
BDT       FIL4   SA4    A4-B2                                                     BDT   300   A
BDT              LX3    B5,X4                                                     BDT   301   A
BDT              MI     X3,FIL3     IF BM[J]                                      BDT   302   A
BDT              BX6    X4-X5       DIFF = XOR(IH,J)                              BDT   303   A
BDT              LX6    -IH.IHP                                                   BDT   304   A
BDT              BX7    -X0*X6                                                    BDT   305   A
BDT              NZ     X7,FIL4     IF IH[DIFF] " 0                               BDT   306   A
BDT              BX3    X4*X5                                                     BDT   307   A
BDT              LX6    IH.IHP-IH.RFP                                             BDT   308   A
BDT              LX3    59-IH.LDP                                                 BDT   309   A
BDT              MI     X3,FIL4     IF LD[IH] & LD[J]                             BDT   310   A
BDT              BX7    -X0*X6                                                    BDT   311   A
BDT              SA3    A4-B1       R1[J]                                         BDT   312   A
BDT              NZ     X7,FIL5     IF RF[DIFF] " 0                               BDT   313   A
BDT              LX6    IH.RFP-IH.CAP                                             BDT   314   A
BDT              BX7    -X0*X6                                                    BDT   315   A
BDT              NZ     X7,FIL4     IF CA[DIFF] " 0                               BDT   316   A
BDT                                                                               BDT   317   A
BDT      *       THE MEMORY REFERENCES INTERFERE, FORM A LINK BETWEEN THEM        BDT   318   A
BDT                                                                               BDT   319   A
BDT       FIL5   IX7    X2+X3                                                     BDT   320   A
BDT              SB3    A7                                                        BDT   321   A
BDT              SA7    A7-1        T = T - 1;  [T] = TREE(EQVB,R1[IH],R1[J])     BDT   322   A
BDT              GE     B3,B6,FIL4  IF STORE ADDR \ LOWER LIMIT                   BDT   323   A
BDT                                                                               BDT   324   A
BDT       FIL6   SA1    O.TREE                  */ CHECK WORKING STORAGE         BDT   325   A
BDT              TREE   FIL                                                       BDT   326   A
BDT              SA2    N.INST                                                    BDT   327   A
BDT              IX1    X1+X2                                                     BDT   328   A
BDT              LX2    1                                                         BDT   329   A
BDT              SA3    N.ST                                                      BDT   330   A
BDT              IX1    X1+X2                                                     BDT   331   A
BDT              IX2    X1+X3       TM = O.TREE + 3*N.INST + N.ST                 BDT   332   A
BDT              SX4    A7-B1                                                     BDT   333   A
BDT              IX1    X2-X4                                                     BDT   334   A
BDT              MI     X1,FIL8     IF TM < T     */ ENOUGH SPACE FOR REST        BDT   335   A
BDT                                                                               BDT   336   A
BDT              EQ     GAS         REALLOCATE                                    BDT   337   A
BDT       FIL8   SPACE  3                                                         BDT   338   A
BDT      **      RESCAN THE *IH* INFO TABLE, SETUP INFORMATION FOR *MCG*          BDT   339   A
BDT      *       SCAN FOR LD/ST COMBINATIONS INTO THE SAME MEMORY LOCATION        BDT   340   A
BDT      *       WHEN THE SECOND HAS THE *SZ* BIT SET, AND SET THE *SR*           BDT   341   A
BDT      *       ( SIZE REDUCABLE ) BIT FOR IT.                                   BDT   342   A
BDT                                                                               BDT   343   A
BDT       FIL8   SA5    B6          I = IHS                                       BDT   344   A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                SX1    B1                                                    BDT          345     A
BDT                MX0    -IH.BMP                                               BDT          346     A
BDT                LX1    D.SRP                                                 BDT          347     A
BDT                SA4    SIOCALL                                               FCC0514      306     A
BDT                SB3    58-D.SZP                                              BDT          348     A
BDT                SB6    A0+B2        BASE = O.TXT + 2                         BDT          349     A
BDT                AX4    1                                                     FCC0514      307     A
BDT                SB4    X4                        (B4) = SIOCALL - 1          FCC0514      308     A
BDT                SB7    A0                        (B7) = TXT                  FCC0514      309     A
BDT                                                                            BDT          350     A
BDT       FIL9     SA5    A5-B2        I = I - 2                                BDT          351     A
BDT                ZR     X5,FIL11     IF [I] = 0                               BDT          352     A
BDT                LX6    B5,X5                                                 BDT          353     A
BDT                MI     X6,FIL9      IF BM[I]                                 BDT          354     A
BDT                                                                            BDT          355     A
BDT                SA3    A5-B1                                                 BDT          356     A
BDT                SA2    X3+B6        DI = BASE + [I-1]                        BDT          357     A
BDT                LX6    B3,X2                                                 BDT          358     A
BDT                PL     X6,FIL9      IF SZ[DI] = 1 */ A 1 PARCEL INSTRCUTION  BDT          359     A
BDT                LX2    58-D.TYP                                              BDT          360     A
BDT                PL     X2,FIL9      IF TYPE[DI] " III  */ NOT A *LD* OR *ST* BDT          361     A
BDT                LX2    2+D.TYP                                               FCC0448      165     A
BDT                                                                            BDT          362         I
          -FCC1126
BDT                SA4    A5+          J = I                                    BDT          363     A
BDT                                                                            BDT          364     A
BDT       FIL10    SA4    A4-B2        J = J - 2                                BDT          365     A
BDT                BX6    X4-X5                                                 BDT          366     A
BDT                LX3    B5,X4                                                 BDT          367     A
BDT                BX7    -X0*X6                                                BDT          368     A
BDT                MI     X3,FIL9      IF BM[J]                                 BDT          369         I
          -FCC0448
BDT                MI     X3,FIL10A    IF BM[J]                                 FCC0448      166         I
          -FCC0514
BDT                MI     X3,FIL10C    IF BM[J]                                 FCC0514      310     A
BDT                NZ     X7,FIL10     IF IHINFO[J] " IHINFO[I]                 BDT          370     A
BDT                                                                            BDT          371     A
BDT       *        IH"S,RF"S AND CA"S MATCH, SET *SR* BIT FOR MASTER MEM REF.  BDT          372     A
BDT                                                                            BDT          373     A
BDT                BX6    X1+X2                                                 BDT          374     A
BDT                SA6    A2           SR[DI] = 1                               BDT          375     A
BDT                ZR     B4,FIL9      IF SIOCALL < 2                           FCC0514      311     A
BDT                LX5    -IH.RFP                                               FCC0514      312     A
BDT                SX7    X5                                                    FCC0514      313     A
BDT                NZ     X7,FIL9      IF RF[I] " 0  */ INDEXED ST              FCC0514      314     A
BDT                                                                            FCC0514      315     A
BDT       *        MASTER IS A REDEF OF J, I MUST BE A *ST*, SAVE *RI* THAT IS FCC0514      316     A
BDT       *        REDEFINED IN THE LINK WORD OF THE *ST* FOR *FTL*.           FCC0514      317     A
BDT                                                                            FCC0514      318     A
BDT                SA3    A4-B1        R = [J-1]                                FCC0514      319     A
BDT                LX4    59-IH.LDP                                             FCC0514      320     A
BDT                MI     X4,FIL10B    IF LD[J]                                 FCC0514      321     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                SA3    B7+X3       R = RI[TXT+R]                                  FCC0514      322     A
BDT        FIL10B  SA4    A2-B2       R1 = DI - 2                                    FCC0514      323     A
BDT                SX6    X3                                                         FCC0514      324     A
BDT                SX5    X4                                                         FCC0514      325     A
BDT                IX7    X5-X6                                                      FCC0514      326     A
BDT                MI     X7,FIL9     IF RI[R1] < R */ ST NOT A FUNCTION OF *LD*     FCC0514      327     A
BDT                                                                                FCC0514      328     A
BDT                SA6    A2+B1       [DI+1] = R    */ SAVE *RI* OF *LD*             FCC0514      329     A
BDT                SA2    A2                                                         FCC1126       14     A
BDT                LX1    D.RDP-D.SRP                                                FCC1126       15     A
BDT                BX6    X1+X2       RD[DI] = 1                                     FCC1126       16     A
BDT                SA6    A2                                                         FCC1126       17     A
BDT                LX1    D.SRP-D.RDP                                                FCC1126       18     A
BDT                SX6    X3                                                         FCC1126       19     A
BDT                                                                                FCC0514      330     A
BDT        *       FORM CHAIN *TST*"S IF THEIR ARE ANY SIMILAR *IP*"S IN THE LOOP     FCC0514      331     A
BDT                                                                                FCC0514      332     A
BDT                AX4    R1.OCP                                                     FCC0514      333     A
BDT                SA3    =XNBIP                                                     FCC0514      334           I
           -CCGNC
BDT                SA3    =XNBIP#                                                    CCGNC        797     A
BDT                SX7    X4-OC.TST-1S10                                             FCC0514      335     A
BDT                NZ     X7,FIL9     IF OC[R1] " OC.TST                             FCC0514      336     A
BDT                ZR     X3,FIL9     IF NBIP = 0   */ NO SIMILAR *IP*"S IN SEQ      FCC0514      337     A
BDT                                                                                FCC0514      338     A
BDT                SA3    TSTC                                                       FCC0514      339     A
BDT                BX6    X3+X6       [DI+1] = TSTC ! [DI+1]  */ ADD TO CHAIN        FCC0514      340     A
BDT                SA6    A6                                                         FCC0514      341           I
           -FCC1126
BDT                SA6    A2+B1                                                      FCC1126       20     A
BDT                SX6    A6                                                         FCC0514      342     A
BDT                LX6    36          TSTC = SHIFT(DI+1,36)  */ CHAIN PTS TO THIS    FCC0514      343     A
BDT                SA6    A3+                                                        FCC0514      344     A
BDT                EQ     FIL9                                                       BDT          376     A
BDT                                                                                FCC0448      167     A
BDT        FIL10A  SA3    A4-B1                                                      FCC0448      168           I
           -FCC0514
BDT        *       CONTINUE SCAN IF J IS A CONDITIONAL JUMP                           FCC0514      345           I
           -CCGA010
BDT        *       CONTINUE SCAN IF J IS A CONDITIONAL JUMP OR *EOS*                  CCGA010        4     A
BDT                                                                                FCC0514      346     A
BDT        FIL10C  SA3    A4-B1                                                      FCC0514      347     A
BDT                ZR     X4,FIL9     IF [J] = 0                                     FCC0448      169     A
BDT                SA3    B6+X3       DJ = BASE + [I-1]                              FCC0448      170     A
BDT                LX3    59-D.TYP                                                   FCC0448      171     A
BDT                PL     X3,FIL10    IF TYPE[DJ] = III  */ JPX OR JIN               FCC0448      172     A
BDT                SA3    A3-B2       R1J = DJ - 2                                   CCGA010        5     A
BDT                AX3    R1.OCP                                                     CCGA010        6     A
BDT                SX6    X3-OC.EOS-2000B                                            CCGA010        7     A
BDT                ZR     X6,FIL10    IF OC[R1J] = OC.EOS                            CCGA010        8     A
BDT                EQ     FIL9                                                       FCC0448      173     A
BDT                                                                                BDT          377     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

BDT       **        SCAN THE EQV LINKS AND UP THE *FT* OF PREDECESSORS TO FORCE          BDT        378    A
BDT       *         THEM TO START BEFORE OTHER COMPUTATIONS THAT WOULD HAVE              BDT        379    A
BDT       *         THE SAME *LST* .                                                     BDT        380    A
BDT                                                                                      BDT        381    A
BDT        FIL11    SA5    A7          TI = T                                            BDT        382    A
BDT                 LX1    D.FTP-D.SRP                                                   BDT        383    A
BDT                 ERRNZ  D.EQVP-D.FTP                                                  FCC0514    348    A
BDT                 ZR     X5,FOL      IF [TI] = 0   */ NO EQUIV LINKS                   BDT        384         I
          -FCC0514
BDT                 ZR     X5,IPL      IF [TI] = 0   */ NO EQUIV LINKS                   FCC0514    349    A
BDT                                                                                      BDT        385    A
BDT        FIL12    SA4    B6+X5       DI = BASE + PRED[TI]                              BDT        386    A
BDT                 SA5    A5+B1       TI = TI + 1                                       BDT        387    A
BDT                 BX6    X4+X1       FT[DI] = FT[DI] ! 1                               BDT        388    A
BDT                 SA6    A4                                                            BDT        389    A
BDT                 NZ     X5,FIL12    IF [TI] " 0                                       BDT        390    A
BDT                                                                                      BDT        391    A
BDT                 SA3    BDTA                                                          BDT        392    A
BDT                 SX6    X3+                                                           BDT        393    A
BDT                 SA6    EQVL        EQVL = BDTA   */ " 0 IF JAM AND EQV LINKS         BDT        394    A
BDT        IPL      EJECT                                                                FCC0514    350    A
BDT       **        IPL - FORM *IP* REDEFINITION LINKS BETWEEN MEMBERS OF A CLASS        FCC0514    351    A
BDT       *         OF SIMILAR *IP*"S, SO ANY REORDERING THAT *RIO* DOES WILL NOT        FCC0514    352    A
BDT       *         CAUSE INCORRECT CODE TO BE GENERATED IF GRA DOES ANY ADDRESS         FCC0514    353    A
BDT       *         DIFFERENCING ( *MFA12* ).                                            FCC0514    354    A
BDT                                                                                      FCC0514    355    A
BDT        IPL      SA1    TSTC                                                          FCC0514    356    A
BDT                 ZR     X1,FOL      IF TSTC = 0   */ NO INCREMENT TST"S               FCC0514    357    A
BDT                                                                                      FCC0514    358         I
          -CCG11
BDT                 SA2    =XO.TET                                                       FCC0514    359    A
BDT                 SA3    =XO.IIT                                                       FCC0514    360    A
BDT                 SA4    O.TREE                                                        FCC0514    361    A
BDT                 SB5    X2                        (B5) = TET                          FCC0514    362    A
BDT                 SB6    X3                        (B6) = IIT                          FCC0514    363    A
BDT                 SB7    X4                        (B7) = TREE                         FCC0514    364    A
BDT                 AX1    36          I = SHIFT(TSTC,-36)                               FCC0514    365    A
BDT                 SB4    A0+B2       BASE = O.TXT + 2                                  FCC1518X    23    A
BDT                 MX0    -D.USESL+1                                                    FCC1518X    24    A
BDT                                                                                      FCC0514    366    A
BDT        IPL1     SA4    X1          LI = I;  LIW = [LI]                               FCC0514    367    A
BDT                 SA3    A4-B2       R2W = [I-2]                                       FCC0514    368    A
BDT                 SA5    A3-B1       R1W = [I-3]                                       FCC0514    369    A
BDT                 LX3    -IH.CAP                                                       FCC0514    370    A
BDT                 SA2    B5+X3       TETW = TET(CA[R2W])                               FCC0514    371    A
BDT                 LX2    -T.ITIP                                                       FCC0514    372    A
BDT                 SX3    X2          J = ITI[TETW]                                     FCC0514    373    A
BDT                 ZR     X3,IPL3     IF J = 0      */INVARIANT TEMP                    FCC0514    374    A
BDT                                                                                      FCC0514    375    A
BDT                 SA3    B6+X3       PIW = IIT(J)                                      FCC0514    376    A
BDT                 LX2    59-T.BIPP+T.ITIP                                              FCC0514    377    A
BDT                 LX3    -PI.CNP     N = CN[PIW]   */ CLASS NUMBER                     FCC0514    378    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                 SX6    X5          R = RI[R1W]   */ RI OF INCREMENT                  FCC0514      379     A
BDT                 PL     X2,IPL2     IF ^BIP[TETW]                                     FCC0514      380     A
BDT                                                                                      FCC0514      381     A
BDT       *         BASE *IP* INCR, ALWAYS BEFORE ^BASE INCR BECAUSE OF *GPO*.           FCC0514      382     A
BDT                                                                                      FCC0514      383     A
BDT                 SA6    B7+X3       TREE(N)  = R  */ SAVE *RI* OF INCR                FCC0514      384     A
BDT                 EQ     IPL3                                                          FCC0514      385     A
BDT                                                                                      FCC0514      386     A
BDT       *         ^BASE INCR, SETUP LINK WORD TO FORM REDEF LINKS FROM USES OF         FCC0514      387     A
BDT       *         *TLD* TO INCREMENT OF BASE IN *FTL* BY SETTING ALTERNATE             FCC0514      388     A
BDT       *         SUCCESSOR FIELD IN LINK WORD IF RLD < RB .                           FCC0514      389          I
          -FCC1518X
BDT       *         SUCCESSOR FIELD IN LINK WORD IF THE *TLD* IS "USED".                 FCC1518X      25     A
BDT                                                                                      FCC0514      390     A
BDT        IPL2     SA2    B7+X3       RB = TREE(N)  */ R-NUM OF BASE INCR               FCC0514      391     A
BDT                 SX2     X2
>>>>      FCCA550X11                                                                     FCCA550X       5     A
BDT                 LX6    T.SUCCP                                                       FCC0514      392     A
BDT                 BX7    X6+X2                                                         FCC0514      393     A
BDT                 SA7    A7-B1       T = T - 1;  [T] = TREE(0,R,RB)                    FCC0514      394     A
BDT                 SX3    X4          RLD = RI[LIW] */ RI OF TLD                        FCC0514      395          I
          -FCC1518X
BDT                 IX7    X3-X2                                                         FCC0514      396          I
          -FCC1518X
BDT                 LX2    18                                                            FCC0514      397          I
          -FCC1518X
BDT                 BX6    X2+X4       LW = SHIFT(RB,18) ! LIW                           FCC0514      398          I
          -FCC1518X
BDT                 AX7    59                                                            FCC0514      399          I
          -FCC1518X
BDT                 BX6    X7*X6       [LI] = IF( RLD < RB ) THEN LW ELSE 0              FCC0514      400          I
          -FCC1518X
BDT                 SA6    A4                                                            FCC0514      401          I
          -FCC1518X
BDT                 SA3    B4+X4       DI = BASE + RI[LIW]  */ DESCR OF *TLD*            FCC1518X      26     A
BDT                 LX2    18                                                            FCC1518X      27     A
BDT                 LX3    -D.USESP-1                                                    FCC1518X      28     A
BDT                 BX7    -X0*X3                                                        FCC1518X      29     A
BDT                 ZR     X7,IPL3     IF USES[DI] = 1  */ ONLY USED IN INCR INST        FCC1518X      30     A
BDT                 BX6    X2+X4       [LI] = [LI] ! SHIFT(RB,18)                        FCC1518X      31     A
BDT                 SA6    A4                                                            FCC1518X      32     A
BDT                                                                                      FCC0514      402     A
BDT        IPL3     AX4    36                                                            FCC0514      403     A
BDT                 SX1    X4          I = SHIFT(LIW,-36)  */ ADVANCE TO NEXT            FCC0514      404     A
BDT                 NZ     X1,IPL1     IF I " 0      */ NOT END OF CHAIN                 FCC0514      405     A
BDT                                                                                      FCC0514      406          I
          -CCG11
BDT                 TRACE  IPL,RLIST                                                     FCC0514      407     A
BDT        FOL      TITLE  FOL - FORM OPERAND LINKS                                      BDT          395     A
BDT       **        FOL - FORM OPERAND LINKS, FORM *BMC* ( INST LINKED TO *EOQ* )        BDT          396     A
BDT                                                                                      BDT          397     A
BDT        FOL      MX0    -R1.RIL+1                                                     BDT          398          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

          -CCG11
BDT                SB4    OC.RS                                                BDT          399          I
          -CCG11
BDT                LX0    1           (X0) = R-NUM EXTRACTION MASK ( A0 => 0 )  BDT          400          I
          -CCG11
BDT        FOL     SX0    RN.MASK     (X0) = R-NUMBER MASK  ( A0 => 0 )         CCG11        131     A
BDT                SB4    OC.RS                                                CCG11        132     A
BDT                SB5    A0+B2       BASE = O.TXT + 2                          BDT          401     A
BDT                SB6    B0          ZPA = 0       */ ZERO PRED ADDRESS        BDT          402     A
BDT                SB7    B0          N.RS = 0      */ N. *RS"S*                BDT          403     A
BDT                SA4    A0+B2       DI = O.TXT + 2                            BDT          404     A
BDT                SX6    A7                                                   BDT          405     A
BDT                SA6    TF          TF = T        */ LWA + 1 OF OPERAND LINKS BDT          406     A
BDT                MX1    D.TYL                                                BDT          407     A
BDT                SA3    O.TREE                                               BDT          408     A
BDT                MX6    0                                                    BDT          409     A
BDT                SA6    X3          BMC = O.TREE;  [BMC] = 0                  BDT          410     A
BDT                SA0    X3                                                   BDT          411     A
BDT                EQ     FOL2                                                 BDT          412     A
BDT                                                                            BDT          413     A
BDT       *        SAVE ADDR OF *SINK* INSTRUCTION ON *BMC* LIST               BDT          414     A
BDT                                                                            BDT          415     A
BDT        FOL1    SX7    A4                                                   BDT          416     A
BDT                PX6    "OC",X7                                              BDT          417     A
BDT                SA6    A0+B1       BMC = BMC + 1;  [BMC] = PK(OC,DI)         BDT          418     A
BDT                SA0    A0+B1                                                BDT          419     A
BDT                                                                            BDT          420     A
BDT       *        GET NEXT INSTRUCTION, AND JUMP TO APPROPIATE PROCESSOR      BDT          421     A
BDT                                                                            BDT          422     A
BDT        FOL2    SA5    A4+B2       R1 = DI + 2                               BDT          423     A
BDT                SA4    A5+B2       DI = R1 + 2                               BDT          424     A
BDT                LX4    58-D.TYP                                             BDT          425     A
BDT                BX7    X1*X4                                                BDT          426     A
BDT                SX6    A4-B5       LA = DI - BASE                            BDT          427     A
BDT                LX6    T.SUCCP                                              BDT          428     A
BDT                NZ     X7,FOL5     IF TYPE[DI] " I                           BDT          429     A
BDT                                                                            BDT          430     A
BDT       *        PROCESS TYPE I INSTRUCTION                                  BDT          431     A
BDT                                                                            BDT          432     A
BDT                LX5    -R1.RKP                                              BDT          433     A
BDT                BX3    -X0*X5                                               BDT          434     A
BDT                LX5    R1.RKP-R1.RJP                                        BDT          435     A
BDT                BX2    -X0*X5                                               BDT          436     A
BDT                LX5    R1.RJP-R1.RIP                                        BDT          437     A
BDT                ZR     X3,FOL3     IF RK[R1] = 0                             BDT          438     A
BDT                IX7    X6+X3                                                BDT          439     A
BDT                SA7    A7-B1       T = T - 1;  [T] = TREE(LA,RK[R1])         BDT          440     A
BDT        FOL3    IX3    X2-X3                                                BDT          441     A
BDT                LX4    1+D.TYP-D.^DP                                        BDT          442     A
BDT                ZR     X2,FOL4     IF RJ[R1] = 0                             BDT          443     A
BDT                ZR     X3,FOL4     IF RJ = RK                                BDT          444     A
BDT                IX7    X6+X2                                                BDT          445     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT              SA7    A7-B1       T = T - 1;  [T] = TREE(LA,RJ[R1])          BDT    446    A
BDT       FOL4   PL     X4,FOL2     IF DEF[DI]                                 BDT    447    A
BDT              UX3    "OC",X5                                                BDT    448    A
BDT              BX5    -X0*X5                                                 BDT    449    A
BDT              IX7    X6+X5                                                  BDT    450    A
BDT              SA7    A7-B1       T = T - 1;  [T] = TREE(LA,RI[R1])          BDT    451    A
BDT              EQ     FOL1                                                   BDT    452    A
BDT                                                                            BDT    453    A
BDT       FOL5   UX3    "OC",X5                                                BDT    454    A
BDT              IX7    X4+X4                                                  BDT    455    A
BDT              MI     X4,FOL8     IF TYPE[DI] = III ! IV                     BDT    456    A
BDT                                                                            BDT    457    A
BDT      **      TYPE II PROCESSING ( S, FMA, CLR , DEF, RS )                  BDT    458    A
BDT                                                                            BDT    459    A
BDT              LX4    1+D.TYP-D.PIP                                          BDT    460    A
BDT              MI     X4,FOL5A    IF PI[DI]     */ RS OR DEF                 BDT    461    A
BDT              SX2    B6                                                     BDT    462    A
BDT              IX7    X6+X2                                                  BDT    463    A
BDT              SA7    A7-B1       T = T - 1;  [T] = TREE(LA,ZPA)             BDT    464    A
BDT              EQ     FOL2                                                   BDT    465    A
BDT                                                                            BDT    466    A
BDT       FOL5A  BX3    -X0*X5                                                 BDT    467    A
BDT              EQ     "OC",B4,FOL6      IF OC[R1] = OC.RS                    BDT    468    A
BDT              SB6    A4-B5       ZPA = LA                                   BDT    469    A
BDT              EQ     FOL1                                                   BDT    470    A
BDT                                                                            BDT    471    A
BDT      *       PROCESS REGISTER STORE                                        BDT    472    A
BDT                                                                            BDT    473    A
BDT       FOL6   LX5    59-R1.INP                                              BDT    474    A
BDT              SB7    B7+B1       N.RS = N.RS + 1                            BDT    475    A
BDT              PL     X5,FOL7     IF IN[R1] = 0                              BDT    476    A
BDT                                                                            BDT    477    A
BDT              SX3    X3-4                                                   BDT    478    A
BDT              IX7    X6+X3                                                  BDT    479    A
BDT              SA7    A7          T = TREE(LA,LA-4) */ ERASE BAD *RJ* LINK   BDT    480    A
BDT              SX6    A7                                                     BDT    481    A
BDT              SA6    A4+B1       [DI+1] = T                                 BDT    482    A
BDT              EQ     FOL1                                                   BDT    483    A
BDT                                                                            BDT    484    A
BDT       FOL7   IX7    X6+X3                                                  BDT    485    A
BDT              SA7    A7-B1       T = T - 1;  [T] = TREE(LA,RI[R1])          BDT    486    A
BDT              SX6    A7                                                     BDT    487    A
BDT              SA6    A4+B1       [DI+1] = T                                 BDT    488    A
BDT              EQ     FOL1                                                   BDT    489    A
BDT                                                                            BDT    490    A
BDT      *       PROCESS TYPE III / IV INSTRUCTIONS                            BDT    491    A
BDT                                                                            BDT    492    A
BDT       FOL8   MI     X7,FOL10    IF TYPE[DI] = IV                           BDT    493    A
BDT                                                                            BDT    494    A
BDT              SA3    A5+B1       R2 = R1 + 1                                BDT    495    A
BDT              LX4    1+D.TYP-D.^DP                                          BDT    496    A
BDT              BX5    -X0*X5      RI = RI[R1]                                BDT    497    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                LX3    -IH.RFP                                          BDT          498     A
BDT                MI     X4,FOL9      IF ^D[DI]                           BDT          499     A
BDT                                                                         BDT          500     A
BDT                BX3    -X0*X3       RN = (RF[R2]/2)*2                   BDT          501     A
BDT                NZ     X3,FOL8A     IF RN " 0                           BDT          502     A
BDT                SX3    B6           RN = ZPA                            BDT          503     A
BDT       FOL8A    IX7    X6+X3                                            BDT          504     A
BDT                SA7    A7-B1        T = T + 1;  [T] = TREE(LA,RN)       BDT          505     A
BDT                EQ     FOL2                                             BDT          506     A
BDT                                                                         BDT          507     A
BDT       FOL9     BX3    -X0*X3                                           BDT          508     A
BDT                IX7    X6+X5                                            BDT          509     A
BDT                LX4    1+D.^DP+58-D.TYP                                 BDT          510     A
BDT                SA7    A7-B1        T = T - 1;  [T] = TREE(LA,RI[R1])   BDT          511     A
BDT                ZR     X3,FOL10     IF RF[R2] = 0                       BDT          512     A
BDT                IX7    X6+X3                                            BDT          513     A
BDT                SA7    A7-B1        T = T - 1;  [T] = TREE(LA,RF[R2])   BDT          514     A
BDT                                                                         BDT          515     A
BDT       FOL10    LX4    1+D.TYP-D.BMP                                    BDT          516     A
BDT                PL     X4,FOL1      IF ^BM[DI]                          BDT          517     A
BDT                SB6    A4-B5        ZPA = LA                            BDT          518     A
BDT                NZ     "OC",FOL1    IF OC[R1] " 0                       BDT          519     A
BDT                                                                         BDT          520     A
BDT                TREE   FOL                                              BDT          521     A
BDT       FTL      TITLE  FTL - FORM TERMINAL ( END OF SEQ ) LINKS         BDT          522     A
BDT      **        FTL - FORM TERMINAL LINKS.                              BDT          523     A
BDT      *         LINK DATA SINKS ( STORES ) AND BOUNDARY MARKERS ( JUMPS, LAB  BDT    524     A
BDT      *         DEFS, ETC ) TO LATER BOUNDARY MARKERS.                  BDT          525     A
BDT      *         FORM REDEF LINKS FOR REG STORES THAT REDEFINE A REGISTER.  BDT       526     A
BDT                                                                         BDT          527     A
BDT                SA1    N.INST                   */ CHECK WORKING STORAGE  BDT          528     A
BDT                SB3    X3                                               BDT          529         I
          -FCC1739X
BDT                SB3    X1                                               FCC1739X       6     A
BDT                SB3    A0+B3                                            BDT          530     A
BDT                SB3    A7-B3                                            BDT          531     A
BDT                SX1    B2-B3                                            BDT          532     A
BDT                PL     X1,GAS       IF BMC + N.INST + 2 \ T             BDT          533     A
BDT                                                                         BDT          534     A
BDT                SA1    A7           TFO = T        */ FWA OF OPERAND LINKS  FCC0514  408     A
BDT                SX1    B6                                               BDT          535     A
BDT                SA5    A0+B1        II = BMC + 1                        BDT          536     A
BDT                LX1    T.SUCCP      LA = ZPA       */ LA = *EOQ* ADDR   BDT          537     A
BDT                SA3    A0                                               BDT          538     A
BDT                SA0    B0           FBM = 0                             BDT          539     A
BDT                BX6    X3                                               BDT          540     A
BDT                SA6    LBM          LBM = [BMC]    */ DI OF LAST *BM*   BDT          541     A
BDT                MX4    23                                               BDT          542     A
BDT                MX6    0                                                BDT          543     A
BDT                ZR     B7,FTL3      IF N.RS = 0                         BDT          544     A
BDT                                                                         BDT          545     A
BDT                SA3    BDTA                                             BDT          546     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                 SA6    REGS                                                  BDT          547     A
BDT       +         LX4    1           FOR I = 0 TO 23;  REGS(I) = 0             BDT          548     A
BDT                 SA6    A6+B1                                                 BDT          549     A
BDT                 MI     X4,*                                                  BDT          550     A
BDT                                                                              BDT          551     A
BDT       *         IF JAM MODE THEN LINK *RI* IN *UJP* RS"S TO FOLLOWING RJX/UJP BDT          552           I
          -FCC0448
BDT       *         SO *SNU* WILL ACCURATELY SET THE *NEXT* USE ADDRESS.          BDT          553           I
          -FCC0448
BDT       *         IF JAM MODE, THEN LINK THE *RI* IN UJP AND FULL LOCK *RS"S*   FCC0448      174     A
BDT       *         TO THE FOLLOWING RJX/EOQ SO *SNU* WILL CORRECTLY SET THE NEXT FCC0448      175     A
BDT       *         USE ADDRESS, IN CASE THE PRED HAS TO BE REISSUED.             FCC0448      176     A
BDT                                                                              BDT          554     A
BDT                 ZR     X3,FTL3     IF BDTA = 0   */ NOT JAM MODE             BDT          555     A
BDT                 SA4    A5          I = II                                    BDT          556     A
BDT                 SB7    OC.JIN                                                BDT          557           I
          -FCC0448
BDT                 BX6    X1          JLA = LA                                  BDT          558           I
          -FCC0448
BDT                 MX2    -SO.LKL                                               BDT          559           I
          -FCC0448
BDT                 BX6    X1          JLA = LA                                  FCC0448      177     A
BDT                 SB7    OC.JIN                                                FCC0448      178     A
BDT                 EQ     FTL1                                                  BDT          560     A
BDT                                                                              BDT          561     A
BDT        FTL0     SB3    X4                                                    BDT          562     A
BDT                 SX6    B3-B5       JLA = [I] - BASE                          BDT          563     A
BDT                 LX6    T.SUCCP                                               BDT          564     A
BDT                                                                              BDT          565     A
BDT        FTL1     SA4    A4-B1       I = I - 1                                 BDT          566     A
BDT                 ZR     X4,FTL3     IF [I] = 0                                BDT          567     A
BDT                 UX5    "OC",X4                                               BDT          568     A
BDT                 GE     "OC",B7,FTL0      IF OC[I] \ OC.JIN                   BDT          569     A
BDT                 NE     "OC",B4,FTL1      IF OC[I] " OC.RS                    BDT          570     A
BDT                                                                              BDT          571     A
BDT                 SA3    X4-2        R1 = [I] - 2                              BDT          572     A
BDT                 BX7    -X0*X3      R = RI[R1]                                BDT          573     A
BDT                 LX3    -R1.SOP-SO.LKP                                        BDT          574           I
          -FCC0448
BDT                 BX5    -X2*X3                                                BDT          575           I
          -FCC0448
BDT                 NZ     X5,FTL1     IF SOLK[R1] " 0  */ NOT A *UJP* LOCK      BDT          576           I
          -FCC0448
BDT                 LX3    59-R1.SOP-SO.LKP                                      FCC0448      179     A
BDT                 MI     X3,FTL1     IF SOLK[R1] = 1!3  */ ^(UJP ! FULL LOCK)  FCC0448      180     A
BDT                 IX7    X6+X7                                                 BDT          577     A
BDT                 SA7    A7-B1       T = T - 1;  [T] = TREE(JLA,R)             BDT          578     A
BDT                 EQ     FTL1                                                  BDT          579     A
BDT        FTL      SPACE  3,14                                                  BDT          580     A
BDT       *         FORM THE BOUNDARY MARKER LINKS                                BDT          581     A
BDT                                                                              BDT          582     A
BDT        FTL2     SX1    A4-B5       LA = DI - BASE                            BDT          583     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                SA0    A4-B5       FBM = LA                                    BDT          584     A
BDT                LX1    T.SUCCP                                                 BDT          585     A
BDT                                                                               BDT          586     A
BDT       FTL3     SA5    A5-B1       II = II - 1                                 BDT          587     A
BDT                SA4    X5          DI = [II]                                   BDT          588     A
BDT                ZR     X5,FTL6     IF DI = 0                                   BDT          589     A
BDT                UX3    "OC",X5                                                 BDT          590     A
BDT                SX6    A4-B5       LINK = DI - BASE                            BDT          591     A
BDT                IX7    X1+X6                                                   BDT          592     A
BDT                LX4    59-D.BMP                                                BDT          593     A
BDT                SA7    A7-B1       T = T - 1;  [T] = TREE(LA,LINK)             BDT          594     A
BDT                MI     X4,FTL2     IF BM[DI]                                   BDT          595     A
BDT                NE     "OC",B4,FTL3      IF OC[R1] " OC.RS                     BDT          596             I
          -FCC0514
BDT                                                                               BDT          597             I
          -FCC0514
BDT                SA2    A4+B1       LW = [DI+1]                                 FCC0514      409     A
BDT                EQ     "OC",B4,FTL3C     IF OC[R1] = OC.RS                     FCC0514      410     A
BDT                ZR     X2,FTL3     IF LW = 0     */ *ST* NOT A REDEF           FCC0514      411             I
          -CCG3
BDT                NZ     X2,FTL3AC   IF LW " 0     */ *ST* IS A REDEF            CCG3         162     A
BDT                LX4    D.BMP-D.ECJP                                            CCG3         163     A
BDT                PL     X4,FTL3     IF ^ECJ[DI]   */ ^ AN RJXJ                  CCG3         164     A
BDT                                                                               CCG3         165     A
BDT       *        RJXJ - FORCE ALL USES OF THE *RF* THAT APPEAR AFTER THE       CCG3         166     A
BDT       *        *RJXJ* TO BE EXECUTED AFTER IT.                               CCG3         167     A
BDT                                                                               CCG3         168     A
BDT                SA3    A4-B1       R2 = DI - 1                                 CCG3         169     A
BDT                SA2    BDTA                                                    CCG3         170     A
BDT                LX3    -IH.RFP                                                 CCG3         171     A
BDT                SB7    X3          R = RF[R2]                                  CCG3         172     A
BDT                SB7    -B7                                                     CCG3         173     A
BDT                LX2    T.RDLP                                                  CCG3         174     A
BDT                BX5    X2+X6                                                   CCG3         175     A
BDT                SA4    A1          TI = TFO                                    CCG3         176     A
BDT                SB6    T.SUCCP                                                 CCG3         177     A
BDT                SA6    EQVL        EQVL = 1                                    CCG3         178     A
BDT                                                                               CCG3         179     A
BDT       FTL3AA   AX7    B6,X4       TL = [TI]                                   CCG3         180     A
BDT                SX2    X4+B7                                                   CCG3         181     A
BDT                SA4    A4+B1       TI = TI + 1                                 CCG3         182     A
BDT                IX3    X6-X7                                                   CCG3         183     A
BDT                PL     X3,FTL3     IF SUCC[TL] @ LINK                          CCG3         184     A
BDT                NZ     X2,FTL3AA   IF PRED[TL] " R                             CCG3         185     A
BDT                LX7    T.SUCCP                                                 CCG3         186     A
BDT                BX7    X7+X5                                                   CCG3         187     A
BDT                SA7    A7-B1       T = T - 1;  [T] = TREE(RDL,SUCC[TL],LINK)   CCG3         188     A
BDT                EQ     FTL3AA                                                  CCG3         189     A
BDT                                                                               FCC0514      412     A
BDT       *        ST IS A REDEFINITION OF PREVIOUS LD/ST, *RD* IS THE R-NUMBER  FCC0514      413     A
BDT       *        BEING REDEFINED.  FORM REDEF LINKS TO FORCE ALL USES OF *RD*  FCC0514      414     A
BDT       *        TO COME BEFORE THIS *ST*.                                     FCC0514      415     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                                                                                  FCC0514      416     A
BDT                 SA3    A4-B2       R1 = DI - 2                                   FCC0514      417          I
         -CCG3
BDT      FTL3AC     SA3    A4-B2       R1 = DI - 2                                   CCG3         190     A
BDT                 SB7    X2          RD = RI[LW]    */ R-NUM BEING REDEF           FCC0514      418     A
BDT                 AX2    18                                                        FCC0514      419     A
BDT                 BX5    -X0*X3      R = RI[R1]     */ SUCCESSOR                   FCC0514      420     A
BDT                 SX2    X2          ARD = SHIFT(LW,-18)  */ BASE INCR R-NUM       FCC0514      421     A
BDT                 SB7    -B7                        (B7) = -RD                     FCC0514      422     A
BDT                 SB6    0           RD2 = 0                                       FCC0514      423     A
BDT                 ZR     X2,FTL3B    IF ARD = 0     */ NO ALTERNATE                FCC0514      424     A
BDT                 SB6    X5          RD2 = R        */ NO LINKS FROM RD2 TO ARD    FCC0514      425     A
BDT                 SX5    X2          R = ARD                                       FCC0514      426     A
BDT                                                                                  FCC0514      427     A
BDT      FTL3B      MX6    0                                                         FCC0514      428     A
BDT                 SA6    A2          [DI+1] = 0     */ CLEAR LINK WORD             FCC0514      429     A
BDT                 SX2    B1                                                        FCC0514      430     A
BDT                 LX5    T.SUCCP                                                   FCC0514      431     A
BDT                 SA4    A1          TI = TFO                                      FCC0514      432     A
BDT                 LX2    T.RDLP                                                    FCC0514      433     A
BDT                 BX6    X5+X2                                                     FCC0514      434     A
BDT                 LX5    -T.SUCCP                                                  FCC0514      435     A
BDT                 SA6    EQVL        EQVL = 1                                      FCC0514      436     A
BDT                                                                                  FCC0514      437     A
BDT      *          SEARCH OPERAND LINKS FOR USES OF *RD* AND FORM REDEF LINKS       FCC0514      438     A
BDT                                                                                  FCC0514      439     A
BDT      FTL3A      BX7    X4          TL = [TI]                                     FCC0514      440     A
BDT                 SX2    X4+B7                                                     FCC0514      441     A
BDT                 AX7    T.SUCCP                                                   FCC0514      442     A
BDT                 SA4    A4+B1       TI = TI + 1                                   FCC0514      443     A
BDT                 SX3    X7+B7                                                     FCC0514      444          I
         -CCGA010
BDT                 ZR     X3,FTL3     IF SUCC[TL] = RD  */ END OF SEARCH            FCC0514      445          I
         -CCGA010
BDT                 SB3    X7+B7                                                     CCGA010        9     A
BDT                 LE     B3,FTL3     IF SUCC[TL] @ RD  */ END OF SEARCH            CCGA010       10     A
BDT                 BX3    X5-X7                                                     FCC0514      446     A
BDT                 NZ     X2,FTL3A    IF PRED[TL] " RD  */ NOT A USE OF *RD*        FCC0514      447     A
BDT                 ZR     X3,FTL3A    IF SUCC[TL] = R  */ DONT FORM LINKS TO SELF   FCC0514      448     A
BDT                 SB3    X7                                                        FCC0514      449     A
BDT                 EQ     B3,B6,FTL3A IF SUCC[TL] = RD2  */ NO LINK BACK TO INCR    FCC0514      450     A
BDT                                                                                  FCC0514      451     A
BDT                 BX7    X6+X7                                                     FCC0514      452     A
BDT                 SA7    A7-B1       T = T - 1;  [T] = TREE(RDL,R,SUCC[TL])        FCC0514      453     A
BDT                 ZR     B6,FTL3A    IF RD2 = 0    */ NO ALTERNATE                 FCC1518X      33     A
BDT                                                                                  FCC1518X      34     A
BDT      *          FORM REDEF LINK FROM NON-BASE INCR TO USE                        FCC1518X      35     A
BDT                                                                                  FCC1518X      36     A
BDT                 LX7    -T.SUCCP                                                  FCC1518X      37     A
BDT                 BX7    X7-X5                                                     FCC1518X      38     A
BDT                 SX2    B6                                                        FCC1518X      39     A
BDT                 BX7    X2+X7                                                     FCC1518X      40     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                LX7    T.SUCCP                                                  FCC1518X     41     A
BDT                SA7    A7-B1       T = T - 1;  [T] = TREE(RDL,RD2,SUCC[TL])     FCC1518X     42     A
BDT                EQ     FTL3A                                                    FCC0514     454     A
BDT       FTL      SPACE  1,10                                                     FCC0514     455     A
BDT      *         REGISTER STORE PROCESSING                                       BDT         598     A
BDT                                                                                BDT         599     A
BDT                SA3    A4-B2       R1 = DI - 2                                  BDT         600         I
         -FCC0514
BDT       FTL3C    SA3    A4-B2       R1 = DI - 2                                  FCC0514     456     A
BDT                BX5    -X0*X3      R = RI[R1]                                   BDT         601     A
BDT                MX2    -SO.REGL                                                 BDT         602     A
BDT                LX3    -R1.SOP                                                  BDT         603     A
BDT                BX7    -X2*X3      REGNO = SOREG[R1]                            BDT         604     A
BDT                LX3    59-SO.LKP                                                BDT         605         I
         -FCC2212X
BDT                PL     X3,FTL3A    IF SOLK[R1] " 1 ! 3  */ ^ TEMP OR RJRS LOCK  BDT         606         I
         -FCC0514  -FCC2212X
BDT                PL     X3,FTL3D    IF SOLK[R1] " 1 ! 3  */ ^ TEMP OR RJRS LOCK  FCC0514     457         I
         -FCC2212X
BDT                SA2    SIOCALL                                                  BDT         607         I
         -FCC2212X
BDT                ZR     X2,FTL3A    IF SIOCALL = 0                               BDT         608         I
         -FCC0514  -FCC2212X
BDT                ZR     X2,FTL3D    IF SIOCALL = 0                               FCC0514     458         I
         -FCC2212X
BDT                                                                                FCC0514     459         I
         -FCC2212X
BDT                SA6    REGS+X7     REGS(REGNO) = LINK                           BDT         609         I
         -FCC2212X
BDT                MX6    0                                                        BDT         610         I
         -FCC2212X
BDT                SA6    A4+B1       [DI+1] = 0                                   BDT         611         I
         -FCC2212X
BDT                LX3    -1                                                       FCC0658       4         I
         -FCC2212X
BDT                PL     X3,FTL3     IF SOLK[R1] " 3                              FCC0658       5         I
         -FCC2212X
BDT                SA2    A7+B1                                                    FCC0658       6         I
         -FCC2212X
BDT                BX7    X2                                                       FCC0658       7         I
         -FCC2212X
BDT                SA7    A7+B1       T = T+1; [T] = [T+1] */ ERASE LINK TO EOQ    FCC0658       8         I
         -FCC2212X
BDT                EQ     FTL3                                                     BDT         612         I
         -FCC2212X
BDT                                                                                BDT         613     A
BDT       FTL3A    SA3    REGS+X7     L = REGS(REGNO) */ ORD OF LAST RS INTO REGNO BDT         614         I
         -FCC0514
BDT       FTL3D    SA3    REGS+X7     L = REGS(REGNO) */ ORD OF LAST RS INTO REGN  FCC0514     460     A
BDT                SA6    A3          REGS(REGNO) = LINK                           BDT         615     A
BDT                SA2    A4+B1       TA = [DI+1]                                  BDT         616     A
BDT                SA4    A4-B1       R2 = DI - 1                                  BDT         617     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT              MX6    0                                                    BDT        618    A
BDT              SA6    A2          [DI+1] = 0                               BDT        619    A
BDT              LX4    -IH.RFP                                              BDT        620    A
BDT              SX4    X4                                                   BDT        621    A
BDT              ZR     X4,FTL4     IF RF[R2] = 0                            BDT        622    A
BDT              BX6    X5          LINK = R                                 BDT        623    A
BDT              SX5    X4          R = RF[R2]                               BDT        624    A
BDT              SA2    TF          TA = TF                                  BDT        625    A
BDT              EQ     FTL4A                                                BDT        626    A
BDT                                                                          BDT        627    A
BDT       FTL4   ZR     X3,FTL3     IF L = 0      */ NO PREVIOUS *RS* INTO REGNO  BDT   628    A
BDT              SX6    X3-4        LINK = L - 4                             BDT        629    A
BDT              LX3    T.SUCCP                                              BDT        630    A
BDT              IX7    X3-X1                                                BDT        631    A
BDT              MI     X7,FTL4A    IF L < LA     */ LAST RS IN THIS BLOCK   BDT        632    A
BDT                                                                          BDT        633    A
BDT              SA4    A4-B1       R1 = R2 - 1                              BDT        634    A
BDT              LX4    59-R1.SOP-SO.LKP                                     BDT        635    A
BDT              MI     X4,FTL3     IF SOLK[R1] = 1 ! 3  */ TEMP OR RJRS LOCK BDT       636    A
BDT                                                                          BDT        637    A
BDT      *       2 *RS*"S INTO THE SAME REGISTER IN A BLOCK, MAKE THE        BDT        638    A
BDT      *       SECOND A REDEFINITION OF THE FIRST.                         BDT        639    A
BDT                                                                          BDT        640    A
BDT       FTL4A  SA2    X2                                                   BDT        641    A
BDT              SA3    BDTA                                                 FCC0448    181    A
BDT              BX5    -X5                                                  BDT        642    A
BDT              LX6    T.SUCCP          (X6) = LINK                         BDT        643    A
BDT              SB7    X5               (B7) = R                            BDT        644    A
BDT              LX3    T.RDLP                                               FCC0448    182    A
BDT              BX6    X6+X3       EQVB = SHIFT(BDTA,T.RDLP)                FCC0448    183    A
BDT              SA6    EQVL        EQVL = 1                                 FCC0448    184    A
BDT       FTL    SPACE  1,10                                                 BDT        645    A
BDT      *       SEARCH FOR USES OF THE R-NUMBER IN INSTRUCTIONS FOLLOWING THE  BDT     646    A
BDT      *       CURRENT *RS*, AND FORM LINKS BETWEEN THEM AND THE INSTRUCTIION  BDT     647    A
BDT      *       PRECEEDING THE SECOND *RS*.                                 BDT        648    A
BDT                                                                          BDT        649    A
BDT       FTL5   SA2    A2-B1       TA = TA - 1                              BDT        650    A
BDT              BX7    X2-X6                                                BDT        651    A
BDT              SX3    X2+B7                                                BDT        652    A
BDT              AX7    T.SUCCP                                              BDT        653    A
BDT              ZR     X7,FTL3     IF SUCC[TA] = LINK  */ END OF SEARCH     BDT        654    A
BDT              NZ     X3,FTL5     IF PRED[TA] " R     */ IF NOT A USE      BDT        655    A
BDT              AX2    T.SUCCP                                              BDT        656    A
BDT              IX7    X6+X2                                                BDT        657    A
BDT              SA7    A7-B1       T = T - 1;  [T] = TREE(LINK,SUCC[TA])    BDT        658         I
         -FCC0448
BDT              SA7    A7-B1       T = T - 1;  [T] = TREE(EQVB,LINK,SUCC[TA])  FCC0448    185    A
BDT              EQ     FTL5                                                 BDT        659    A
BDT                                                                          BDT        660    A
BDT       FTL6   SB6    A0-4                                                 BDT        661         I
         -FCC0514
BDT              SA4    N.INST                                               BDT        662         I

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

          -FCC0514
BDT                NZ     B6,FTL7     IF FBM " 4    */ NO LABEL AT *BOS*                BDT          663          I
          -FCC0514
BDT        FTL6    SA4    N.INST                                                        FCC0514      461     A
BDT                BX7    X1                                                            BDT          664     A
BDT                SA7    A7-B1       T = T - 1;  [T] = TREE(FBM,0)                     BDT          665     A
BDT                                                                                     BDT          666     A
BDT        FTL7    SB6    X4                        */ CHECK WORKING STORAGE            BDT          667          I
          -FCC0514
BDT                SB6    X4                        */ CHECK WORKING STORAGE            FCC0514      462     A
BDT                SB6    A5+B6                                                         BDT          668     A
BDT                SB7    A7          FTREE = T                                         BDT          669     A
BDT                SX1    B6-B7                                                         BDT          670     A
BDT                GE     B6,B7,GAS   IF O.TREE + N.INST \ T                            BDT          671     A
BDT                TREE   FTL                                                           BDT          672     A
BDT                                                                                     BDT          673     A
BDT       *        SORT THE TEMPORARY TREE TO GROUP EDGES BY SUCCESSOR                  BDT          674     A
BDT                                                                                     BDT          675     A
BDT                SA2    TREEL                                                         BDT          676     A
BDT                SA3    EQVL                                                          BDT          677     A
BDT                SX6    A0                                                            BDT          678     A
BDT                SX7    A7                                                            BDT          679     A
BDT                SA6    FBM                                                           BDT          680     A
BDT                SA7    FTREE       FTREE = T                                         BDT          681     A
BDT                IX1    X2-X7       LEN = TREEL - FTREE                               BDT          682     A
BDT                NZ     X3,FTL8     IF EQVL " 0                                       BDT          683     A
BDT                CALL   SHL         SORT( TREE , [SUCC,PRED] )                        BDT          684          I
          -CCG1
BDT                CALL   SHL=        SORT( TREE , [SUCC,PRED] )                        CCG1         289          I
          -CCGNC
BDT                CALL   SHL#        SORT( TREE , [SUCC,PRED] )                        CCGNC        798     A
BDT                EQ     FTL10                                                         BDT          685     A
BDT                                                                                     BDT          686     A
BDT       *        JAM MODE AND EQUIV LINKS, USE MASK SORT                              BDT          687     A
BDT                                                                                     BDT          688     A
BDT        FTL8    MX0    -1                                                            BDT          689          I
          -FCC0448
BDT        FTL8    MX0    -2                                                            FCC0448      186     A
BDT                SB6    B0                                                            BDT          690     A
BDT                MX5    0                                                             BDT          691     A
BDT                LX0    T.EQVP                                                        BDT          692     A
BDT                CALL   SST         SORT( TREE , [SUCC,PRED] )                        BDT          693          I
          -CCG1
BDT                CALL   SST=        SORT( TREE , [SUCC,PRED] )                        CCG1         290          I
          -CCGNC
BDT                CALL   SST#        SORT( TREE , [SUCC,PRED] )                        CCGNC        799     A
BDT                                                                                     FCC0514      463     A
BDT        FTL10   BSS    0                                                             FCC0514      464     A
BDT        FJL     TITLE  FJL - FORM JUMP LINKS                                         BDT          694     A
BDT        FTL10   BSS    0                                                             BDT          695          I
          -FCC0514
BDT                                                                                     BDT          696          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

                -FCC0514
      BDT       **        FJL - FORM LINKS TO JUMPS FROM BELOW TO CONSTRAIN INSTRUCTIIONS        BDT          697     A
      BDT       *         TO BE TIED TO THE BLOCK THEY APPEAR IN.                                BDT          698     A
      BDT                                                                                        BDT          699     A
      BDT                 SA1    FTREE                                                           BDT          700     A
      BDT                 SA3    O.TREE                                                          BDT          701     A
      BDT                 SA4    FBM                                                             BDT          702     A
      BDT                 SA5    L.TXT                                                           BDT          703     A
      BDT                 SX6    X4+8                                                            BDT          704     A
      BDT                 MX7    0                                                               BDT          705     A
      BDT                 IX6    X6-X5                                                           BDT          706     A
      BDT                 ZR     X4,FJL      IF FBM = 0    */ NO BOUNDARY MARKERS                BDT          707     A
      BDT                 NZ     X6,FJL1     IF FBM+8 < L.TXT  */ SEQ HAS AN INTERIOR JP          BDT          708     A
      BDT                                                                                        BDT          709     A
      BDT       *         NO JUMPS                                                               BDT          710     A
      BDT                                                                                        BDT          711     A
      BDT        FJL      BX6    X1                                                              BDT          712     A
      BDT                 SA6    A3          O.TREE = FTREE                                      BDT          713     A
      BDT                 SA7    X1-1        [FTREE-1] = 0                                       BDT          714     A
      BDT                 EQ     FJL6                                                            BDT          715     A
      BDT                                                                                        BDT          716     A
      BDT       *         MOVE THE TEMP TREE DOWN AND ADD JUMP LINKS WHERE NECESSARY             BDT          717     A
      BDT       *         ADD *JP* LINK IF ALL PREDECESSORS LIE BELOW THE *JP*                   BDT          718     A
      BDT                                                                                        BDT          719     A
      BDT        FJL1     SA5    X1-1        TI = FTREE - 1                                      BDT          720     A
      BDT                 SA2    O.TXT                                                           BDT          721     A
      BDT                 MX0    -T.SUCCL                                                        BDT          722     A
      BDT                 SB7    X4          JPL = FBM                                           BDT          723     A
      BDT                 SB5    X2+2        BASE = O.TXT + 2                                    BDT          724     A
      BDT                 SB6    60-T.SUCCP                                                      BDT          725     A
      BDT                 LX0    T.SUCCP                                                         BDT          726     A
      BDT                 SA7    X3          TS = O.TREE;  [TS] = 0                              BDT          727     A
      BDT                 SX6    X3+B1                                                           BDT          728     A
      BDT                 SA6    A3          O.TREE = O.TREE + 1                                 BDT          729     A
      BDT                                                                                        BDT          730     A
      BDT       *         MOVE LINKS PRECEEDING THE FIRST *BM* DOWN                              BDT          731     A
      BDT                                                                                        BDT          732     A
      BDT        FJL2     SA5    A5+B1       TI = TI + 1                                         BDT          733     A
      BDT                 LX4    B6,X5                                                           BDT          734     A
      BDT                 BX7    X5                                                              BDT          735     A
      BDT                 SB3    X4                                                              BDT          736     A
      BDT                 SA7    A7+B1       TS = TS + 1;  [TS] = [TI]                           BDT          737     A
      BDT                 LE     B3,B7,FJL2  IF SUCC[TI] @ JPL                                   BDT          738     A
      BDT                                                                                        BDT          739     A
      BDT       *         SCAN THE SET OF EDGES FOR EACH INSTRUCTION ( GROUPED BY SUCC )         BDT          740     A
      BDT       *         AND IF THE LAST PRED ( MAX ) LINKS TO A NODE BELOW THE                 BDT          741           I
                -FCC0715A
      BDT       *         AND IF THE MAX OF THE PRED < R, LINK TO A NODE BELOW THE               FCC0715A       4     A
      BDT       *         LAST ENCOUNTERED *BM* , FORM A LINK TO IT.                             BDT          742     A
      BDT                                                                                        BDT          743     A
      BDT        FJL3     BX6    -X0*X5      R = SUCC[TI]                                        BDT          744     A
      BDT                 LX4    B6,X5                                                           BDT          745     A

                0        1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                SB3    X4+                                                  BDT          746          I
          -FCC0715A
BDT                                                                            BDT          747          I
          -FCC0715A
BDT       FJL4     SB4    X5          M = PRED[TI]                             BDT          748          I
          -FCC0715A
BDT                SA5    A5+B1       TI = TI + 1                              BDT          749          I
          -FCC0715A
BDT                SB3    X4                                                   FCC0715A       5     A
BDT                SB4    B0          M = 0                                    FCC0715A       6     A
BDT                                                                            FCC0715A       7     A
BDT       FJL4     SB2    X5                                                   FCC0715A       8     A
BDT                GT     B2,B3,FJL4A IF PRED[TI] > R  */ A REDEFINITION LINK FCC0715A       9     A
BDT                SB4    X5          M = PRED[TI]                             FCC0715A      10     A
BDT       FJL4A    SA5    A5+1        TI = TI + 1                              FCC0715A      11     A
BDT                BX4    X6-X5                                                BDT          750     A
BDT                LX7    X5                                                   BDT          751     A
BDT                BX3    -X0*X4                                               BDT          752     A
BDT                SA7    A7+B1       TS = TS + 1;  [TS] = [TI]                BDT          753     A
BDT                ZR     X3,FJL4     IF SUCC[TI] = R                          BDT          754     A
BDT                                                                            BDT          755     A
BDT                SX3    B7+                                                  BDT          756     A
BDT                SA2    B5+B3       DI = BASE + R                            BDT          757     A
BDT                IX6    X6+X3                                                BDT          758     A
BDT                GE     B4,B7,FJL5  IF M \ JPL                               BDT          759     A
BDT                SA6    A7          [TS] = TREE(R,JPL)                       BDT          760     A
BDT                SA7    A7+B1       TS = TS + 1;  [TS] = [TI]                BDT          761     A
BDT                                                                            BDT          762     A
BDT       FJL5     LX2    59-D.BMP                                             BDT          763     A
BDT                PL     X2,FJL3     IF ^BM[DI]                               BDT          764     A
BDT                SB7    B3          JPL = R                                  BDT          765     A
BDT                NZ     X5,FJL3     IF [TI] " 0                              BDT          766     A
BDT                                                                            BDT          767     A
BDT                SX6    A7          TREEL = TS                               BDT          768     A
BDT                SA6    TREEL                                                BDT          769     A
BDT                                                                            BDT          770     A
BDT                IF     DEF,/DEBUG/FJL,1                                     BDT          771     A
BDT                SNAP   -O.TREE,-TREEL,,1,1000,1,TREE-FJL                    BDT          772          I
          -FEAT155F
BDT       TREE-FJL SNAP   *O.TREEL,*TREEL                                      FEAT155F     263          I
          -CCG11
BDT       TREE-FJL SNAP   *O.TREE,TREEL                                        CCG11        133     A
BDT                                                                            BDT          773     A
BDT       FJL6     SA1    SIOCALL                                              BDT          774     A
BDT                ZR     X1,CIP      IF SIOCALL = 0                           BDT          775     A
BDT                                                                            BDT          776     A
BDT       *        BDT CALLED FROM *RIO*, SET *TREE* LENGTH AND EXIT           BDT          777     A
BDT                                                                            BDT          778     A
BDT                SA2    O.TREE                                               BDT          779     A
BDT                SA3    TREEL                                                BDT          780     A
BDT                IX6    X3-X2                                                BDT          781     A
BDT                SA6    L.TREE      L.TREE = TREEL - O.TREE                  BDT          782     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                MX7    0                                                    BDT          783     A
BDT                SA7    A1          SIOCALL = 0                              BDT          784     A
BDT                EQ     BDT                                                  BDT          785     A
BDT       CIP      TITLE  CIP - CALCULATE INSTRUCTION PRIORITIES               BDT          786     A
BDT      **        CIP - CALCULATE INSTRUCTION PRIORITIES                      BDT          787     A
BDT      *         CALCULATES TIME TO TRAVEL FROM *EOQ* TO *BOS* OF NETWORK    BDT          788     A
BDT      *         THROUGH THE NODES. NODE TIME IS EITHER THE INSTRUCTION TIME BDT          789     A
BDT      *         FUN UNIT ) OR 1 ( FOR A NODE DEPTH CALCULATION ).           BDT          790     A
BDT      *         THEN CHANGE THE THE "PRIORITIES" INTO "LATE START TIMES",   BDT          791     A
BDT      *         WHERE  LST[LI] = SEQTIME - PRIOR[LI] , IS THE LATEST TIME   BDT          792     A
BDT      *         THAT AN INSTRUCTION MAY START WITHOUT LENGTHING THE SCHEDULE. BDT        793     A
BDT                                                                            BDT          794     A
BDT       CIP      MX0    -D.FTL                                               BDT          795     A
BDT                SA1    BDTA                                                 BDT          796     A
BDT                LX0    D.FTP                                                BDT          797     A
BDT                SA4    O.TXT                                                BDT          798     A
BDT                SB4    I.PRIORP-D.FTP                                       BDT          799     A
BDT                SA0    X4+3        BASE = O.TXT + 3                         BDT          800     A
BDT                NZ     X1,CIP6     IF BDTA " 0   */ JAM MODE                BDT          801     A
BDT                SA5    N.ST                                                 CCG0207        5     A
BDT                AX5    6                                                    CCG0207        6     A
BDT                ZR     X5,CIP1                     */ IF N.ST .LT. 64       CCG0207        7     A
BDT                SA1    A5                                                   CCG0207        8     A
BDT                SA3    N.INST                                               CCG0207        9     A
BDT                SB3    X1                                                   CCG0207       10     A
BDT                AX3    1                                                    CCG0207       11     A
BDT                SB5    X3                                                   CCG0207       12     A
BDT                GT     B3,B5,CIP6        */IF N.INST/2 .GT. N.ST, JAM MODE  CCG0207       13     A
BDT                                                                            BDT          802     A
BDT       CIP1     SA3    TREEL                                                BDT          803     A
BDT                SA7    A1+         LBM = JPA                                BDT          804     A
BDT                SA5    X3-1        TI = TREEL - 1                           BDT          805     A
BDT                MX1    -D.USESL                                             BDT          806     A
BDT                LX1    D.USESP                                              BDT          807     A
BDT                SB7    60-T.SUCCP                                           BDT          808     A
BDT                SB6    X5          P = PRED[TI]                             BDT          809     A
BDT                LX5    -T.SUCCP                                             BDT          810     A
BDT                SB5    X5          R = SUCC[TI]                             BDT          811     A
BDT                SB3    A5          TL = TI                                  BDT          812     A
BDT                SA4    A0+B5       PI = BASE + R                            BDT          813     A
BDT                SA3    A0+B6       PTI = BASE + P                           BDT          814     A
BDT                SA2    A4-B1       DI = PI - 1   */ FT[*EOQ*] = 1           BDT          815     A
BDT                                                                            BDT          816     A
BDT      *         CALCULATE TIME TO REACH THIS INSTRUCTION                    BDT          817     A
BDT                                                                            BDT          818     A
BDT       CIP2     BX6    -X0*X2                                               BDT          819     A
BDT                LX5    B4,X6                                                BDT          820     A
BDT                IX7    X4+X5       TOTIME = PRIOR[PI] + FT[DI]              BDT          821     A
BDT                BX6    -X1*X2                                               BDT          822     A
BDT                                                                            BDT          823     A
BDT      *         PROPAGATE TIME TO REACH THIS INSTRUCTION TO ITS PREDECESSORS BDT         824     A
BDT                                                                            BDT          825     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT       CIP3     SA5    A5-B1       TI = TI - 1                                  BDT          826     A
BDT                IX4    X3-X7                                                    BDT          827     A
BDT                SB6    X5          P = PRED[TI]                                 BDT          828     A
BDT                LX3    B7,X5                                                    BDT          829     A
BDT                SB2    X3                                                       BDT          830     A
BDT                PL     X4,CIP4     PRIOR[PTI] = MAX( PRIOR[PTI] , TOTIME )      BDT          831     A
BDT                SA7    A3                                                       BDT          832     A
BDT       CIP4     SA3    A0+B6       PTI = BASE + P                               BDT          833     A
BDT                EQ     B2,B5,CIP3  IF SUCC[TI] = R                              BDT          834     A
BDT                                                                                BDT          835     A
BDT       *        SET PRIORITY INFO IN LAST WORD OF INSTRUCTION                   BDT          836     A
BDT                                                                                BDT          837     A
BDT                LX6    I.USESP-D.USESP                                          BDT          838     A
BDT                SX4    A5-B3       NPRED[PI] = -( TI - TL )                     BDT          839     A
BDT                BX7    X7+X6       USES[PI] = USES[DI]                          BDT          840     A
BDT                LX4    I.NPREDP                                                 BDT          841     A
BDT                IX6    X7-X4                                                    BDT          842     A
BDT                SA6    A4                                                       BDT          843     A
BDT                                                                                BDT          844     A
BDT                SB5    B2          R = SUCC[TI]                                 BDT          845     A
BDT                SA4    A0+B2       PI = BASE + R                                BDT          846     A
BDT                SB3    A5          TL = TI                                      BDT          847     A
BDT                SA2    A4-B1       DI = PI - 1                                  BDT          848     A
BDT                NZ     X5,CIP2     IF [TI] " 0                                  BDT          849     A
BDT                BX7    X4                                                       BDT          850     A
BDT                AX7    I.PRIORP                                                 BDT          851     A
BDT                SA7    SEQTIME     SEQTIME = PRIOR[PI]  */ TIME TO REACH *BOS*  BDT          852     A
BDT                                                                                BDT          853     A
BDT       *        CONVERT THE PRIORITIES TO "LATE START TIMES".                   BDT          854         I
          -FCC0641
BDT       *        CONVERT THE PRIORITIES TO *LATE START TIMES*, SET THE *LST"S*   FCC0641          78     A
BDT       *        OF *RS* INSTRUCTIONS TO ZERO TO FORCE THEM TO BE ISSUED         FCC0641          79     A
BDT       *        IMMEDIATELY AFTER THEIR PREDECESSORS IN *MCG*.                  FCC0641          80     A
BDT                                                                                BDT          855     A
BDT                MX0    -I.PRIORL                                                BDT          856     A
BDT                LX7    I.PRIORP                                                 BDT          857     A
BDT                SA1    N.INST                                                   BDT          858     A
BDT                SB4    4                                                        BDT          859     A
BDT                SB2    X1-1        I = N - 1                                    BDT          860     A
BDT                SA4    A4+B4       LI = BASE + 4                                BDT          861     A
BDT                LX0    I.PRIORP                                                 BDT          862     A
BDT                BX5    -X0*X4                                                   BDT          863         I
          -FCC0641
BDT                                                                                BDT          864         I
          -FCC0641
BDT       CIP5     IX2    X7-X5       LST[LI] = SEQTIME - PRIOR[LI]                BDT          865         I
          -FCC0641
BDT                BX3    X0*X4                                                    BDT          866         I
          -FCC0641
BDT                SA4    A4+B4       LI = LI + 4                                  BDT          867         I
          -FCC0641
BDT                IX6    X2+X3                                                    BDT          868         I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

          -FCC0641
BDT                 SB2    B2-1        I = I - 1                                  BDT          869          I
          -FCC0641
BDT                 BX5    -X0*X4                                                 BDT          870          I
          -FCC0641
BDT                 SA6    A4-B4                                                  BDT          871          I
          -FCC0641
BDT                 NZ     B2,CIP5     IF I " 0                                   BDT          872          I
          -FCC0641
BDT                 SB5    OC.RS                                                  FCC0641       81     A
BDT                 BX5    -X0*X4      PR = PRIOR[LI]                             FCC0641       82     A
BDT                                                                               FCC0641       83     A
BDT       CIP5      IX2    X7-X5       LST[LI] = SEQTIME - PR                     FCC0641       84     A
BDT                 SA1    A4+B1       R1 = LI + 1   */ R1 WORD OF NEXT           FCC0641       85     A
BDT                 BX3    X0*X4                                                  FCC0641       86     A
BDT                 SA4    A4+B4       LI = LI + 4                                FCC0641       87     A
BDT                 IX6    X2+X3                                                  FCC0641       88     A
BDT                 SB2    B2-B1       I = I - 1                                  FCC0641       89     A
BDT                 UX1    B3,X1                                                  FCC0641       90     A
BDT                 BX5    -X0*X4      PR = PRIOR[LI]                             FCC0641       91     A
BDT                 SA6    A4-B4                                                  FCC0641       92     A
BDT                 NE     B3,B5,CIP5A IF OC[R1] " OC.RS                          FCC0641       93     A
BDT                 BX5    X7          PR = SEQTIME  */ FORCE LST = 0             FCC0641       94     A
BDT       CIP5A     NZ     B2,CIP5     IF I " 0                                   FCC0641       95     A
BDT                                                                               BDT          873     A
BDT       DBG       IF     DEF,/DEBUG/TREE                                        BDT          874     A
BDT                 SNAPRL BDT-X                                                  BDT          875     A
BDT                 SA1    O.TREE                                                 BDT          876     A
BDT                 SA5    TREEL                                                  BDT          877     A
BDT                 IX6    X5-X1                                                  BDT          878     A
BDT                 BX7    X1                                                     BDT          879     A
BDT                 SA6    L.TREE      L.TREE = TREEL - O.TREE                    BDT          880     A
BDT                 SA7    DTSA+1                                                 BDT          881     A
BDT                 SX6    A0-3                                                   BDT          882     A
BDT                 SA6    A7+2                                                   BDT          883     A
BDT                 SA1    DTSA                                                   BDT          884     A
BDT                 RJ     =XDMPTREE   SNAP THE DEPENDENCY TREE                   BDT          885     A
BDT                 RJ     =XRESTORE                                              BDT          886          I
          -FEAT155F
BDT                 RJ     =XRESET=                                               FEAT155F     264          I
          -CCG11
BDT                 RJ     RSR=                                                   CCG11        134     A
BDT       DBG       ENDIF                                                         BDT          887     A
BDT                                                                               BDT          888     A
BDT                 EQ     SIT                                                    BDT          889     A
BDT       CIP-JAM   EJECT                                                         BDT          890     A
BDT       **        JAM MODE "PRIORITY" CALCULATION                               BDT          891     A
BDT       *         ELIMINATE DUPLICATE EDGES FROM THE TREE. ( FOR MCG/RIL )      BDT          892     A
BDT       *         REFORMAT IT AS A USES INDEX TABLE.                            BDT          893     A
BDT       *         SETUP THE USES, LST AND NPRED FIELDS IN THE LINK WORD.        BDT          894     A
BDT                                                                               BDT          895     A
BDT       CIP6      SA3    O.TREE                                                 BDT          896     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                MX1    -D.USESL                                                BDT          897     A
BDT                SX0    B1+B1       PR = 2                                      BDT          898     A
BDT                LX1    D.USESP                                                 BDT          899     A
BDT                SB7    X3          TF = O.TREE                                 BDT          900     A
BDT                MX7    0                                                       BDT          901     A
BDT                LX0    U.LSTP                                                  BDT          902     A
BDT                SA7    B7-B1       UI = O.TREE - 1;  [UI] = 0 */ STORE ADDR    BDT          903     A
BDT                SB6    60-T.SUCCP                                              BDT          904     A
BDT                SA5    X3          TI = O.TREE   */ FETCH ADDR                 BDT          905     A
BDT                SB5    B7-B1       UF = O.TREE - 1                             BDT          906     A
BDT                SB2    I.LSTP-U.LSTP                                           BDT          907     A
BDT                BX2    X0                                                      BDT          908     A
BDT                SB3    A0-B1                                                   BDT          909     A
BDT                                                                               BDT          910     A
BDT       *        COUNT PREDECESSORS OF CURRENT SUCCESSOR                        BDT          911     A
BDT                                                                               BDT          912     A
BDT        CIP7    BX6    X5          TL = [TI]                                   BDT          913     A
BDT                SA5    A5+B1       TI = TI + 1                                 BDT          914     A
BDT                LX7    B6,X6                                                   BDT          915     A
BDT                BX4    X6-X5                                                   BDT          916     A
BDT                ZR     X4,CIP7     IF [TI] = TL  */ DUP EDGE                   BDT          917     A
BDT                                                                               BDT          918     A
BDT                IX7    X0+X7                                                   BDT          919     A
BDT                AX4    T.SUCCP                                                 BDT          920     A
BDT                SA7    A7+B1       UI = UI + 1;  [UI] = SIT(PRED[TL],PR,SUCC[TL])  BDT          921     A
BDT                ZR     X4,CIP7     IF SUCC[TI] = SUCC[TL]                      BDT          922     A
BDT                                                                               BDT          923     A
BDT       *        SETUP INFO IN LINK WORD                                        BDT          924     A
BDT                                                                               BDT          925     A
BDT                SA3    B3+X7       DI = O.TXT+2 + SUCC[TL]                     BDT          926     A
BDT                SX4    A7-B5       NPRED[LW] = UI - UF                         BDT          927     A
BDT                LX7    B2,X0       LST[LW] = PR                                BDT          928     A
BDT                LX4    I.NPREDP                                                BDT          929     A
BDT                BX3    -X1*X3                                                  BDT          930     A
BDT                IX0    X0+X2       PR = PR + 2                                 BDT          931     A
BDT                LX3    I.USESP-D.USESP   USES[LW] = USES[DI]                   BDT          932     A
BDT                BX4    X4+X7                                                   BDT          933     A
BDT                SB5    A7          UF = UI                                     BDT          934     A
BDT                IX6    X4+X3                                                   BDT          935     A
BDT                SA6    A3+B1       LW = DI + 1                                 BDT          936     A
BDT                NZ     X5,CIP7     IF [TI] " 0                                 BDT          937     A
BDT                                                                               BDT          938     A
BDT                BX6    X0                                                      BDT          939     A
BDT                SA6    A0          LST[BASE] = PR                              BDT          940     A
BDT                MX7    0                                                       BDT          941     A
BDT                SA7    A7+B1       UI = UI + 1;  [UI] = 0                      BDT          942     A
BDT                SX6    A7                                                      BDT          943     A
BDT                SA6    TREEL       TREEL = UI                                  BDT          944     A
BDT                SX1    A7-B7       L.TREE = UI - O.TREE                        BDT          945     A
BDT                BX7    X1                                                      BDT          946     A
BDT                SA3    EQVL                                                    BDT          947     A
BDT                ZR     X3,SIT1A    IF EQVL = 0                                 BDT          948     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                                                                    BDT          949     A
BDT                MX0    -1                                           BDT          950          I
          -FCC0448
BDT                MX0    -2                                           FCC0448      187     A
BDT                SB6    B0                                           BDT          951     A
BDT                MX5    0                                            BDT          952     A
BDT                LX0    U.EQVP                                       BDT          953     A
BDT                SA7    L.TREE                                       BDT          954     A
BDT                CALL   SST         SORT( TREE , [PRED,SUCC] )       BDT          955          I
          -CCG1
BDT                CALL   SST=        SORT( TREE , [PRED,SUCC] )       CCG1         291          I
          -CCGNC
BDT                CALL   SST#        SORT( TREE , [PRED,SUCC] )       CCGNC        800     A
BDT                EQ     SIT1B                                        BDT          956     A
BDT        SIT     SPACE  2           SIT                              BDT          957     A
BDT       **       SIT - REFORMAT THE TREE AS A *USES INDEX* TABLE.    BDT          958     A
BDT       *                24/PRED,18/LST[SUCC],18/SUCC                 BDT          959     A
BDT                                                                    BDT          960     A
BDT        SIT     SA1    TREEL                                        BDT          961     A
BDT                SA5    X1-1        UI = TREEL - 1                   BDT          962     A
BDT                MX0    -I.LSTL                                      BDT          963     A
BDT                SB4    A0          BASE = O.TXT + 3                 BDT          964     A
BDT                LX0    I.LSTP                                       BDT          965     A
BDT                SB5    60-T.SUCCP                                   BDT          966     A
BDT                LX6    B5,X5                                        BDT          967     A
BDT                                                                    BDT          968     A
BDT        SIT1    SA4    B4+X6       PW = BASE + SUCC[UI]             BDT          969     A
BDT                SA5    A5-B1       UI = UI - 1                      BDT          970     A
BDT                BX3    -X0*X4                                       BDT          971     A
BDT                LX3    U.LSTP-I.LSTP                                BDT          972     A
BDT                IX7    X3+X6       LST[UI+1] = LST[PI]              BDT          973     A
BDT                LX6    B5,X5                                        BDT          974     A
BDT                SA7    A5+B1                                        BDT          975     A
BDT                NZ     X5,SIT1     IF [UI] " 0                      BDT          976     A
BDT                                                                    BDT          977     A
BDT                SX6    A5+B1                                        BDT          978     A
BDT                SB7    A5+B1       O.TREE = UI + 1                  BDT          979     A
BDT                IX7    X1-X6       L.TREE = TREEL - O.TREE          BDT          980     A
BDT                BX1    X7                                           BDT          981     A
BDT        SIT1A   SA7    L.TREE                                       BDT          982     A
BDT                CALL   SHL         SORT( TREE , [PRED,LST[SUCC],SUCC] )  BDT     983          I
          -CCG1
BDT                CALL   SHL=        SORT( TREE , [PRED,LST[SUCC],SUCC] )  CCG1    292          I
          -CCGNC
BDT                CALL   SHL#        SORT( TREE , [PRED,LST[SUCC],SUCC] )  CCGNC   801     A
BDT                                                                    CCGNC        802          I
          -CCG11
BDT        SIT     SPACE  3                                            BDT          984          I
          -FCC1126
BDT        SIT     SPACE  2                                            FCC1126       21     A
BDT       *        NOW SETUP THE USES INDEX INFO IN THE LINK WORD OF THE INST  BDT  985     A
BDT                                                                    BDT          986     A

          0        1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT       SIT1B    MX0    1                                                BDT        987    A
BDT                SB3    U.PREDP                                          BDT        988    A
BDT                LX0    1+U.LASTP                                        BDT        989    A
BDT                SA5    B7          UI = O.TREE                          BDT        990    A
BDT                MX1    -U.PREDL                                         BDT        991    A
BDT                SB6    B7          UF = O.TREE                          BDT        992    A
BDT                LX1    U.PREDP                                          BDT        993    A
BDT                SB5    A0          BASE = O.TXT + 3                     BDT        994    A
BDT                EQ     SIT3                                             BDT        995    A
BDT                                                                        BDT        996    A
BDT       SIT2     BX6    X2+X3       [LW] = UINFO(USES,INDX,LST,NPRED)    BDT        997    A
BDT                SA6    A3+                                              BDT        998    A
BDT                                                                        BDT        999    A
BDT       SIT3     LX4    X5          UL = UI                              BDT       1000    A
BDT                SA5    A5+B1       UI = UI + 1                          BDT       1001    A
BDT                BX6    X4-X5                                            BDT       1002    A
BDT                BX7    -X1*X6                                           BDT       1003    A
BDT                ZR     X7,SIT3     IF PRED[UI] = PRED[UL]               BDT       1004    A
BDT                                                                        BDT       1005    A
BDT                AX6    B3,X4                                            BDT       1006    A
BDT                SX2    B6-B7       INDX = UF - O.TREE                   BDT       1007    A
BDT                BX7    X0+X4       LAST[UI-1] = 1                       BDT       1008    A
BDT                SA3    B5+X6       LW = BASE + PRED[UL]                 BDT       1009    A
BDT                LX2    I.INDXP                                          BDT       1010    A
BDT                SB6    A5          UF = UI                              BDT       1011    A
BDT                SA7    A5-B1                                            BDT       1012    A
BDT                NZ     X5,SIT2     IF [UI] " 0                          BDT       1013    A
BDT                BX6    X2+X3                                            BDT       1014    A
BDT                SA6    A3                                               BDT       1015    A
BDT                                                                        BDT       1016    A
BDT        DBG     IF     DEF,/DEBUG/BDT                                   BDT       1017    A
BDT                SNAPRL BDT-X                                            BDT       1018    A
BDT                SA2    O.TREE                                           BDT       1019    A
BDT                SX7    A0-3                                             BDT       1020    A
BDT                BX6    X2                                               BDT       1021    A
BDT                SA6    DTSA+1                                           BDT       1022    A
BDT                SA7    A6+2                                             BDT       1023    A
BDT                SA1    DTSA                                             BDT       1024    A
BDT                CALL   DMPSIT                                           BDT       1025    A
BDT                CALL   RESET=                                           BDT       1026          I
          -CCG11
BDT                RJ     RSR=                                             CCG11      135    A
BDT        DBG     ENDIF                                                   BDT       1027    A
BDT                EQ     BDT                                              BDT       1028    A
BDT                                                                        BDT       1029    A
BDT        DBG     IF     DEF,DEBUG                                        BDT       1030          I
          -FCC1056
BDT                USE    SNAPS                                            BDT       1031          I
          -FCC1056
BDT        .T      IFNE   TEST,0                                           FCC1056     46    A
BDT                USE    DEBUG                                            FCC1056     47    A
BDT        DTSA    CON    DTSB,0,L.TREE                                    BDT       1032    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                BSSZ   2                                                    BDT        1033     A
BDT       DTSB     BSSZ   1          SNAP COUNT                                BDT        1034     A
BDT                USE    0                                                    BDT        1035     A
BDT       DBG      ENDIF                                                       BDT        1036          I
         -FCC1056
BDT       .T       ENDIF                                                       FCC1056      48     A
BDT       GAS      SPACE  3          GAS                                       BDT        1037     A
BDT      **        GAS - GET ADDITIONAL SPACE TO BUILD THE TREE AND RESTART    BDT        1038     A
BDT      *         NOTE - THIS SUBROUTINE SHOULD ALMOST NEVER BE CALLED IF THE BDT        1039     A
BDT      *         INITIAL STORAGE REQUEST IS SUFFICIENT.                      BDT        1040     A
BDT      *         ENTRY  (X1) = MIN SPACE NEEDED                              BDT        1041     A
BDT      *         EXIT   *TREE* SPACE INCREASED, POINTERS ADJUSTED, TO *FIL*  BDT        1042     A
BDT                                                                            BDT        1043     A
BDT       GAS      SX1    X1+40B                                               BDT        1044     A
BDT                ALLOC  TREE,X1    GET SPACE                                 BDT        1045     A
BDT                SA5    O.TXT                                                BDT        1046     A
BDT                SX3    X3-1                                                 BDT        1047     A
BDT                IX6    X2+X3                                                BDT        1048     A
BDT                SA6    TREEL                                                BDT        1049     A
BDT                SA0    X5         (A0) = O.TXT                              BDT        1050     A
BDT                SB2    B1+B1      (B2) = 2                                  BDT        1051     A
BDT                SB4    B2+B2      (B4) = 4                                  BDT        1052     A
BDT                EQ     FIL                                                  BDT        1053     A
BDT                                                                            BDT        1054     A
BDT                QUAL                                                        BDT        1055     A
BDT       RIO      TITLE  RIO - RESET INSTRUCTION ORDER                        BDT        1056     A
BDT      *IF       DEF,RIO,1                                                   CCG1        293     A
BDT                TRACER (RIO,SIO,CRW)                                        CCG1        294     A
BDT      **        RIO - RESET INSTRUCTION ORDER                               BDT        1057     A
BDT      *                                                                     BDT        1058     A
BDT      *         RIO REORDERS THE INSTRUCTIONS TO BRING THEM TO AN ORDER THAT BDT        1059     A
BDT      *         IS "MINIMUM" WIDTH WITH RESPECT TO ITS X-REGISTER           BDT        1060     A
BDT      *         REQUIREMENTS.  THE ALGORITHM USED HERE IS STRAIGHT FROM     BDT        1061     A
BDT      *         AHO-ULLMAN VOLUME II, AND ACHIEVES A MINIMUM ONLY IN THE    BDT        1062     A
BDT      *         CASE OF NO MULTIPLE USES.                                   BDT        1063     A
BDT      *                                                                     BDT        1064     A
BDT      *         ENTRY  SEQUENCE TO BE REORDERED IN *TXT*, LINK WORDS ARE ZERO, BDT        1065     A
BDT      *                USES COUNTS AND PRECEDENCE BITS SET IN DESCRIPTORS. BDT        1066     A
BDT      *                (X6) = RMODE, 0 IF CALL FROM MCG, 1 IF CALL FROM GPO FCC0514      465     A
BDT      *                                                                     BDT        1067     A
BDT      *         EXIT   (X6) = MAX REG WIDTH OF THE SEQUENCE                 BDT        1068     A
BDT      *                *REGW* FIELD SET IN THE DESCRIPTORS OF ALL INSTRUCTIONS. BDT        1069     A
BDT      *                LINK WORDS CLEARED.                                 BDT        1070     A
BDT                                                                            BDT        1071     A
BDT      *         JT. - RIO TREE AFTER REFORMATTING                           BDT        1072     A
BDT                                                                            BDT        1073     A
BDT                DESCRIBE JT.,60    JT(SUCC,LVL,PRED)                        BDT        1074     A
BDT       SUCC     DEFINE 18                                                   BDT        1075     A
BDT                DEFINE 6                                                    BDT        1076     A
BDT       LVL      DEFINE 18          LEVEL OF PRED  ( NODE LABEL OR WEIGHT )  BDT        1077     A
BDT       PRED     DEFINE 18                                                   BDT        1078     A
BDT                                                                            BDT        1079     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

BDT       *         I. - FORMAT OF LINK WORD DURING DEPTH FIRST SEARCH ( *SIO* )          BDT        1080    A
BDT                                                                                      BDT        1081    A
BDT                 DESCRIBE I.,60     LW(FLAGS,PRI,LVL,NP)                              BDT        1082    A
BDT        MI       DEFINE 1           MARKED INSTRCUTION ( VISITED )                    BDT        1083    A
BDT        II       DEFINE 1           ISSUED INSTRUCTION                                BDT        1084    A
BDT                 DEFINE 4                                                             BDT        1085        I
          -FCC0791
BDT        RDST     DEFINE 1           =1 IF A REDEF ST BELOW INST IN THE STACK          FCC0791       5    A
BDT                 DEFINE 3                                                             FCC0791       6        I
          -FCC1126
BDT        STCP     DEFINE 1           =1 IF ST CHAIN PROCESSED                          FCC1126      22    A
BDT                 DEFINE 2                                                             FCC1126      23    A
BDT        PRI      DEFINE 18          ADDRESS OF PRED LIST IN *TREE*                    BDT        1086    A
BDT        LVL      DEFINE 18          LEVEL                                             BDT        1087        I
          -FCC0514
BDT        LVL      DEFINE 18          LEVEL / ST CHAIN POINTER                          FCC0514     466    A
BDT        NP       DEFINE 18          N.PRED NOT YET VISITED                            BDT        1088    A
BDT                                                                                      BDT        1089    A
BDT        OI       DEQU   NP          OUTPUT INDEX ( FINAL ORED OF INSTRUCTION )        BDT        1090    A
BDT                                                                                      FCC1290       6    A
BDT        MIN.PR   EQU    10B         MIN PRIORITY FOR NON-TERMINAL INSTRUCTIONS        FCC1290       7    A
BDT        RIO      SPACE  2                                                             BDT        1091    A
BDT        RIO      ENTRY. **                                                            BDT        1092        I
          -CCGNC
BDT        RIO      ENTRY. **,#                                                          CCGNC       803    A
BDT                 SA6    RMODE                                                         BDT        1093        I
          -FCC0514
BDT                 SX7    B1                                                            BDT        1094        I
          -FCC0514
BDT                 NZ     X6,RIO1     IF RMODE " 0  */ CALL FROM *GPO*                  FCC0514     467    A
BDT                 SA1    =XPSLC                                                        FCC0514     468        I
          -FCC0675
BDT                 NZ     X1,SIO10    IF PSLC " 0   */ OUTSIDE OF ALL LOOPS             FCC0514     469        I
          -FCC0675
BDT                 SA1    =XHBI                                                         FCC0675     256        I
          -CCGNC
BDT                 SA1    =XHBI#                                                        CCGNC       804    A
BDT                 ZR     X1,SIO10    IF HBI = 0    */ OUTSIDE OF ALL LOOPS             FCC0675     257    A
BDT                                                                                      FCC0514     470    A
BDT        RIO1     SA6    RMODE                                                         FCC0514     471    A
BDT                 SX7    X6+B1                                                         FCC0514     472    A
BDT                 MX6    0                                                             BDT        1095    A
BDT                 SA7    SIOCALL                                                       BDT        1096    A
BDT                 CALL   BDT         FORM THE DEPENDENCY GRAPH                         BDT        1097    A
BDT                                                                                      BDT        1098    A
BDT       *         ALLOCATE EXTRA SPACE IN *TREE* FOR THE STACK, AND MOVED INSTS.       BDT        1099    A
BDT                                                                                      BDT        1100    A
BDT                 SA3    N.INST                                                        BDT        1101    A
BDT                 SA4    L.TREE                                                        BDT        1102    A
BDT                 SA5    L.TXT                                                         BDT        1103    A
BDT                 SB3    X3+B1                                                         BDT        1104    A
BDT                 IX1    X5-X4                                                         BDT        1105    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                SB4    X1+B1                                                  BDT        1106    A
BDT                GT     B4,B3,RIO2  L = MAX( N.INST+2 , L.TXT - L.TREE )       BDT        1107    A
BDT                SX1    B3+B1                                                  BDT        1108    A
BDT       RIO2     ALLOC  TREE,X1     ALLOC( TREE , L )                          BDT        1109    A
BDT                MX6    0                                                      BDT        1110    A
BDT                SA0    X2                          (A0) = TREE                BDT        1111    A
BDT                SA6    X2+B6       TREE(OLD(LTREE)) = 0                       BDT        1112    A
BDT       RIO      SPACE  2,14                                                   BDT        1113    A
BDT      *         REFORMAT THE TREE, CALCULATE THE INSTRUCTION NODE LABELS AS   BDT        1114    A
BDT      *         PER AHO-ULLMAN, SETUP LINK WORDS FOR DEPTH FIRST SEARCH.      BDT        1115    A
BDT                                                                              BDT        1116    A
BDT                MX7    2                                                      BDT        1117    A
BDT                SA4    O.TXT                                                  BDT        1118    A
BDT                SX6    B1                                                     BDT        1119         I
         -FCC1056
BDT                SX6    B1+B1                                                  FCC1056      49         I
         -FCC1290
BDT                SB2    MIN.PR                      (B2) = MIN.PR              FCC1290       8    A
BDT                SX6    B2                                                     FCC1290       9    A
BDT                SB7    X4+3        TB = TXT + 3                               BDT        1120    A
BDT                LX6    I.LVLP                                                 BDT        1121    A
BDT                BX7    X7+X6                                                  BDT        1122    A
BDT                SA7    B7          [TB] = LW(II,MI,0,1,0)                     BDT        1123         I
         -FCC1056
BDT                SA7    B7          [TB] = LW(II,7I,0,2,0)                     FCC1056      50         I
         -FCC1290
BDT                SA7    B7          [TB] = LW(II,MI,0,MIN.PR,0)                FCC1290      10    A
BDT                MX0    -T.SUCCL                                               BDT        1124    A
BDT                SB6    A0          TF = TREE                                  BDT        1125    A
BDT                LX0    T.SUCCP                                                BDT        1126    A
BDT                MX1    0           DLV = 0       */ DIFFERENCE OF LEVELS      BDT        1127    A
BDT                SX2    B1          LLV = 1       */ LAST *LV*                 BDT        1128         I
         -FCC1056
BDT                SB4    B1          MLV = 1       */ MAX LVL OF PRED           BDT        1129         I
         -FCC1056
BDT                SX2    B1+B1       LLV = 2       */ LAST *LV*                 FCC1056      51         I
         -FCC1290
BDT                SB4    B1+B1       MLV = 2       */ MAX LVL OF PRED           FCC1056      52         I
         -FCC1290
BDT                SX2    B2          LLV = MIN.PR  */ LAST *LV*                 FCC1290      11    A
BDT                SB4    B2          MLV = MIN.PR  */ MAX LVL OF PRED           FCC1290      12    A
BDT                SA5    A0          TI = TREE                                  BDT        1130    A
BDT                SB5    B7-B1       DB = TB - 1                                BDT        1131    A
BDT                EQ     RIO5                                                   BDT        1132    A
BDT                                                                              BDT        1133    A
BDT      *         STORE REFORMATTED EDGE, ADVANCE TO NEXT EDGE                   BDT        1134    A
BDT                                                                              BDT        1135    A
BDT       RIO4     SA7    A5          [TI] = TN                                  BDT        1136    A
BDT                BX4    X5          TL = [TI]                                  BDT        1137    A
BDT                SB4    B3          MLV = LV                                   BDT        1138    A
BDT                SA5    A5+B1       TI = TI + 1                                BDT        1139    A
BDT                BX6    X4-X5                                                  BDT        1140    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                BX7    -X0*X6                                          BDT        1141     A
BDT                NZ     X7,RIO6     IF SUCC[TI] " SUCC[TL]              BDT        1142     A
BDT                                                                       BDT        1143     A
BDT      *         REFORMAT EDGE, UPDATE DLV, MLV                         BDT        1144     A
BDT                                                                       BDT        1145     A
BDT       RIO5     LX5    -T.PREDP    R = PRED[TI]                        BDT        1146     A
BDT                SA3    B7+X5       LI = TB + R                         BDT        1147     A
BDT                LX3    -I.LVLP     LV = LVL[LI]                        BDT        1148     A
BDT                BX2    X2-X3                                           BDT        1149     A
BDT                BX1    X1+X2       DLV = DLV ! XOR(LLV,LV)             BDT        1150     A
BDT                SB3    X3                                              BDT        1151     A
BDT                SX2    X3          LLV = LV                            BDT        1152     A
BDT                SX6    X5                                              BDT        1153     A
BDT                LX5    T.PREDP                                         BDT        1154     A
BDT                LX6    JT.PREDP                                        BDT        1155     A
BDT                BX4    -X0*X5                                          BDT        1156     A
BDT                LX4    JT.SUCCP-T.SUCCP                                BDT        1157     A
BDT                SX3    X3                                              BDT        1158     A
BDT                BX6    X4+X6                                           BDT        1159     A
BDT                LX3    JT.LVLP                                         BDT        1160     A
BDT                BX7    X6+X3       TN = JT(SUCC[TI],LV,R)              BDT        1161     A
BDT                GE     B3,B4,RIO4  IF LV \ MLV                         BDT        1162     A
BDT                                                                       BDT        1163     A
BDT      *         LV < MLV , SWAP EDGES                                  BDT        1164     A
BDT                                                                       BDT        1165     A
BDT                SA4    A5-B1       TLL = [TI-1]                        BDT        1166     A
BDT                SB3    B4+         LV = MLV                            BDT        1167     A
BDT                SA7    A4          [TI-1] = TN                         BDT        1168     A
BDT                BX7    X4          TN = TLL                            BDT        1169     A
BDT                EQ     RIO4                                            BDT        1170     A
BDT                                                                       BDT        1171     A
BDT      *         TERMINATE EDGE GROUP, INITIALIZE FOR NEW GROUP         BDT        1172     A
BDT                                                                       BDT        1173     A
BDT       RIO6     SX6    B6          PRI = TF                            BDT        1174     A
BDT                SX3    A5-B6       NP = TI - TF  */ N.PRED             BDT        1175     A
BDT                LX6    I.PRIP                                          BDT        1176     A
BDT                LX3    I.NPP                                           BDT        1177     A
BDT                SX1    X1                                              BDT        1178     A
BDT                BX7    X6+X3                                           BDT        1179     A
BDT                LX4    -T.SUCCP    S = SUCC[TI]                        BDT        1180     A
BDT                NZ     X1,RIO7     IF DLV " 0    */ PRED AT DIFF LEVELS BDT       1181     A
BDT                AX3    I.NPP+1                                         BDT        1182     A
BDT                ZR     X3,RIO7     IF NP = 1                           BDT        1183     A
BDT                SB4    B4+B1       MLV = MLV + 1                       BDT        1184     A
BDT                                                                       BDT        1185     A
BDT      *         IF INSTRUCTION *PRS*, THEN CHECK FOR A TEMP LOCK, AND IF SO,  BDT 1186          I
          -FCC1290
BDT      *         SET *LVL* TO ZERO TO FORCE IT TO COME JUST BEFORE IS USES  BDT 1187          I
          -FCC1290
BDT      *         ( ILS,IRS KLUDGE ).                                    BDT        1188          I
          -FCC1290
BDT      *         IF INSTRUCTION IS TERMINAL ( ST OR *PRS* ) THEN ADJUST *LVL*  FCC1290   13   A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT      *         SO THE ORDER OF THE INSTRUCTIONS WILL BE - ST"S, PRS TO X0,6,7          FCC1290      14     A
BDT      *         AND PRS TO A LD-REG. IF THE INSTRUCTION PRS TO A B-REG AND IT           FCC1290      15     A
BDT      *         IS A TEMP LOCK, THEN SET *LVL* TO 0 TO FORCE IT TO COME                 FCC1290      16     A
BDT      *         JUST BEFORE ITS USES. THIS IS THE INDEXED SHIFT KLUDGE.                 FCC1290      17     A
BDT                                                                                        BDT        1189     A
BDT       RIO7     SA3    B5+X4       DI = DB + S                                          BDT        1190     A
BDT                SX2    B4+                                                              BDT        1191     A
BDT                LX3    59-D.PRSP                                                        BDT        1192     A
BDT                PL     X3,RIO7A    IF ^PRS[DI]                                          BDT        1193          I
         -FCC0625
BDT                SA3    A3+2        R1RS = DI + 2                                        BDT        1194          I
         -FCC0514  -FCC0625
BDT                LX3    D.PRSP-D.LDP                                                     FCC0514     473          I
         -FCC0625
BDT                PL     X3,RIO7B    IF ^LD[DI]                                           FCC0514     474          I
         -FCC0625
BDT                SA3    A3-B1       R2 = DI - 1                                          FCC0514     475          I
         -FCC0625
BDT                LX3    59-IH.LDP                                                        FCC0514     476          I
         -FCC0625
BDT                AX3    59                                                               FCC0514     477          I
         -FCC0625
BDT                BX2    -X3*X2      MLV = ^LD[IH] & MLV  */ MLV = 0 IF APLIST L          FCC0514     478          I
         -FCC0625
BDT                EQ     RIO7A                                                            FCC0514     479          I
         -FCC0625
BDT                                                                                        FCC0514     480          I
         -FCC0625
BDT       RIO7B    SA3    A3+2        R1RS = DI + 2                                        FCC0514     481          I
         -FCC0625
BDT                LX3    59-R1.SOP-SO.LKP                                                 BDT        1195          I
         -FCC0625
BDT                PL     X3,RIO7A    IF SOLK[R1RS] " 1  */ ^ TEMP LOCK                    BDT        1196          I
         -FCC0625
BDT                PL     X3,RIO7A    IF ^PRS[DI]                                          FCC0625      19          I
         -FCC0881
BDT                SA3    A3+2        R1RS = DI + 2                                        FCC0625      20          I
         -FCC0881
BDT                LX3    58-R1.SOP-SO.LKP                                                 FCC0625      21          I
         -FCC0881
BDT                MI     X2,RIO7A    IF SOLK[R1RS] > 1  */ ^ UJP OR TEMP LOCK             FCC0625      22          I
         -FCC0881
BDT                SX2    B0          MLV = 0       */ FORCE LATE OUTPUT IN *SIO*          BDT        1197          I
         -FCC0881
BDT                                                                                        BDT        1198          I
         -FCC0881
BDT       RIO7A    LX2    I.LVLP                                                           BDT        1199          I
         -FCC0881
BDT                PL     X3,RIO7B    IF ^PRS[DI]                                          FCC0881     238          I
         -FCC1056
BDT                SA1    A3+2        R1RS = DI + 2                                        FCC0881     239          I
         -FCC1056

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                MI     X3,RIO7AA   IF PRS[DI]                                  FCC1056      53     A
BDT                LX3    D.PRSP-D.STP                                            FCC1056      54     A
BDT                PL     X3,RIO7B    IF ^ST[DI]                                  FCC1056      55     A
BDT                                                                               FCC1056      56     A
BDT      *         FORCE STORES TO COME BEFORE RS"S TO X-REGISTERS                FCC1056      57     A
BDT                                                                               FCC1056      58     A
BDT                GT     B4,B1,RIO7B MLV = MAX( MLV , 2 )                        FCC1056      59          I
         -FCC1290
BDT                SX2    B1+B1                                                   FCC1056      60          I
         -FCC1290
BDT                GE     B4,B2,RIO7B MLV = MAX( MLV, MIN.PR )                    FCC1290      18     A
BDT                SX2    B2                                                      FCC1290      19     A
BDT                EQ     RIO7B                                                   FCC1056      61     A
BDT                                                                               FCC1056      62     A
BDT       RIO7AA   SA1    A3+2        R1RS = R1 + 2                               FCC1056      63          I
         -FCC1322
BDT       RIO7AA   SA1    A3+2        R1RS = DI + 2                               FCC1322       5          I
         -FCC1825X
BDT       RIO7AA   LX3    D.PRSP-D.ZPP                                            FCC1825X     25     A
BDT                SX2    B0          SAVE = MLV;  MLV = 0  */ FORCE LATE OUTPUT  FCC1825X     26     A
BDT                MI     X3,RIO7B    IF ZP[DI]                                   FCC1825X     27     A
BDT                SA1    A3+2        R1RS = DI + 2                               FCC1825X     28     A
BDT                LX1    58-R1.SOP-SO.LKP                                        FCC0881     240          I
         -FCC1290
BDT                PL     X1,RIO7A    IF SOLK[R1RS] < 2  */ UJP ! TEMP LOCK       FCC0881     241          I
         -FCC1290
BDT                LX1    SO.LKP-SO.RTP                                           FCC0881     242          I
         -FCC1290
BDT                PL     X1,RIO7B    IF SORT[R1RS] " 2  */ ^RS TO AN X-REG       FCC0881     243          I
         -FCC1290
BDT                LX1    58-R1.SOP-SO.RTP                                        FCC1290      20     A
BDT                PL     X1,RIO7A    IF SORT[R1RS] " 2  */ ^RS TO AN X-REGISTER  FCC1290      21     A
BDT                                                                               FCC0881     244     A
BDT                LX1    2+SO.RTP-SO.RNP                                         FCC0881     245     A
BDT                MX3    -SO.RNL                                                 FCC0881     246     A
BDT                BX1    -X3*X1                                                  FCC0881     247     A
BDT                SB2    X1                                                      FCC0881     248          I
         -FCC1290
BDT                SX3    301B                                                    FCC0881     249          I
         -FCC1290
BDT                AX3    B2,X3                                                   FCC0881     250          I
         -FCC1290
BDT                SB3    X1                                                      FCC1290      22     A
BDT                SX3    301B                                                    FCC1290      23     A
BDT                AX3    B3,X3                                                   FCC1290      24     A
BDT                MX1    -1                                                      FCC0881     251     A
BDT                BX2    -X1*X3      MLV = 1 & SHIFT(301B,-SORN[R1RS])           FCC0881     252     A
BDT                SX2    X2+2        MLV = MLV + 2                               FCC1290      25     A
BDT                NZ     B3,RIO7B    IF NOT X0                                   CCG28        43     A
BDT                SX2    X2+1              MLV = MLV + 1                         CCG28        44     A
BDT                EQ     RIO7B                                                   FCC0881     253     A
BDT                                                                               FCC0881     254     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT       RIO7A    SX2    0           MLV = 0       */ FORCE LATE OUTPUT IN *SIO*         FCC0881      255          I
         -FCC1290
BDT       RIO7A    LX1    SO.RTP-SO.LKP                                                    FCC1290       26     A
BDT                MI     X1,RIO7B    IF SOLK[R1RS] = 2 */ ^(TEMP ! UJP) LOCK              FCC1290       27          I
         -FCC1322
BDT                SX2    B0          MLV = 0       */ FORCE LATE OUTPUT IN *SIO*         FCC1290       28          I
         -FCC1322
BDT                SX2    B0          SAVE = MLV;  MLV = 0 */ FORCE LATE OUTPUT           FCC1322        6          I
         -FCC1825X
BDT                PL     X1,RIO7B    IF SOLK[R1RS] < 2  */ IF TEMP OR UJP LOCK           FCC1322        7     A
BDT                LX3    D.PRSP-D.ZPP                                                     FCC1322        8          I
         -FCC1825X
BDT                MI     X3,RIO7B    IF ZP[DI]                                            FCC1322        9          I
         -FCC1825X
BDT                SX2    B4          MLV = SAVE                                           FCC1322       10     A
BDT                                                                                        FCC0881      256     A
BDT       RIO7B    LX2    I.LVLP                                                           FCC0881      257     A
BDT                BX6    X2+X7                                                            BDT         1200     A
BDT                SA6    B7+X4       [TB+S] = LW(0,PRI,MLV,NP)                            BDT         1201     A
BDT                LX5    -T.PREDP    R = PRED[TI]                                         BDT         1202     A
BDT                SA2    B7+X5                                                            BDT         1203     A
BDT                SB6    A5          TF = TI                                              BDT         1204     A
BDT                LX2    -I.LVLP     LLV = LVL[TB+R]                                      BDT         1205     A
BDT                MX1    0           DLV = 0                                              BDT         1206     A
BDT                SB4    X2          MLV = LLV                                            BDT         1207     A
BDT                LX5    T.PREDP                                                          BDT         1208     A
BDT                NZ     X5,RIO5     IF [TI] " 0   */ NOT END OF TREE                     BDT         1209     A
BDT       RIO      SPACE  2,14                                                             BDT         1210          I
         -FCC1290
BDT      *         SEARCH FOR TERMINAL *RS*"S TO STORE REGISTERS AND CLEAR THE             BDT         1211          I
         -FCC1290
BDT      *         *LVL* IN THE TREE ENTRY SO THAT THEY MAY BE ISSUED LAST.                BDT         1212          I
         -FCC1290
BDT                                                                                        BDT         1213          I
         -FCC1290
BDT                SB7    B7-3                                                             BDT         1214          I
         -FCC1290
BDT                SB4    OC.RS                                                            BDT         1215          I
         -FCC1290
BDT                LX4    JT.SUCCP                                                         BDT         1216          I
         -FCC1290
BDT                LX0    JT.SUCCP-T.SUCCP                                                 BDT         1217          I
         -FCC1290
BDT                SA3    A5          TJ = TI                                              BDT         1218          I
         -FCC1290
BDT                MX5    -JT.LVLL                                                         BDT         1219          I
         -FCC1290
BDT                MX7    -SO.REGL                                                         BDT         1220          I
         -FCC1290
BDT                LX5    JT.LVLP                                                          BDT         1221          I
         -FCC1290
BDT                                                                                        BDT         1222          I

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
          -FCC1290
BDT        RIO8     SA3    A3-B1       TJ = TJ - 1                                    BDT        1223          I
          -FCC1290
BDT                 BX6    X3-X4                                                      BDT        1224          I
          -FCC1290
BDT                 LX3    -JT.PREDP                                                  BDT        1225          I
          -FCC1290
BDT                 SA2    B7+X3       R1 = TXT(PRED[TJ])                             BDT        1226          I
          -FCC1290
BDT                 BX6    -X0*X6                                                     BDT        1227          I
          -FCC1290
BDT                 NZ     X6,RIO9     IF SUCC[TJ] " S                                BDT        1228          I
          -FCC1290
BDT                 UX1    B2,X2                                                      BDT        1229          I
          -FCC1290
BDT                 LX2    -R1.SOP                                                    BDT        1230          I
          -FCC1290
BDT                 NE     B2,B4,RIO8  IF OC[R1] " OC[RS                              BDT        1231          I
          -FCC1290
BDT                                                                                   BDT        1232          I
          -FCC1290
BDT                 BX6    -X7*X2                                                     BDT        1233          I
          -FCC1290
BDT                 SX1    X6-26B                                                     BDT        1234          I
          -FCC1290
BDT                 MI     X1,RIO8     IF REG[R1] < 26B  */ NOT A STORE REG           BDT        1235          I
          -FCC1290
BDT                 LX3    JT.PREDP                                                   BDT        1236          I
          -FCC1290
BDT                 BX6    X5*X3       LVL[TJ] = 0                                    BDT        1237          I
          -FCC1290
BDT                 SA6    A3                                                         BDT        1238          I
          -FCC1290
BDT                 EQ     RIO8                                                       BDT        1239          I
          -FCC1290
BDT        RIO      SPACE  1,10                                                       BDT        1240     A
BDT       *         DETERMINE THE LIMITS OF THE TREE THAT HAVE TO BE SORTED, AND      BDT        1241     A
BDT       *         SORT THAT SUBSET ( USUALLY THE NULL SET ).                        BDT        1242     A
BDT                                                                                   BDT        1243     A
BDT        RIO9     SA1    A0          TL = [TREE]                                    BDT        1244          I
          -FCC1290
BDT                 SA1    A0          TL = [TREE]                                    FCC1290      29     A
BDT                 SA2    A0+B1       TJ = TREE + 1                                  BDT        1245     A
BDT                 SB7    B0          FWA = 0       */ FWA OF SORT SET               BDT        1246     A
BDT                 SB2    B1+B1                                                      BDT        1247     A
BDT                                                                                   BDT        1248     A
BDT        RIO10    IX3    X2-X1       DIF = [TJ] - TL                                BDT        1249     A
BDT                 BX1    X2          TL = [TJ]                                      BDT        1250     A
BDT                 SA2    A2+1        TJ = TJ + 1                                    BDT        1251     A
BDT                 PL     X3,RIO10    IF DIF \ 0    */ ELEMENTS IN ORDER             BDT        1252     A
BDT                                                                                   BDT        1253     A
BDT                 SB6    A2-1        LWA = TJ - 1                                   BDT        1254     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                ZR     X1,RIO11     IF TL = 0      */ END OF TREE                    BDT        1255     A
BDT                NZ     B7,RIO10     IF FWA " 0     */ NOT FIRST REVERSAL             BDT        1256     A
BDT                SB7    A2-B2        FWA = TJ - 2                                     BDT        1257     A
BDT                EQ     RIO10                                                         BDT        1258     A
BDT                                                                                    BDT        1259     A
BDT       RIO11    BSS    0                                                             BDT        1260     A
BDT                IF     DEF,/DEBUG/RIO,1                                              BDT        1261     A
BDT                SNAP   -A0,-A5,,1,100,1,TREE-RIO                                     BDT        1262          I
          -FEAT155F
BDT       TREE-RIO SNAP   *A0,*A5                                                       FEAT155F    265     A
BDT                TRACE  RIO,RLIST                                                     BDT        1263     A
BDT                                                                                    BDT        1264     A
BDT                SA0    B6           ST = LWA       */ STACK ADDR FOR SIO             BDT        1265     A
BDT                ZR     B7,RIO12     IF FWA = 0     */ NO REVERSALS                   BDT        1266     A
BDT                                                                                    FCC0514     482     A
BDT                SA2    B7           T = [FWA]                                        FCC0514     483     A
BDT                SA3    O.TXT                                                         FCC0514     484     A
BDT                SB5    X3+3                                                          FCC0514     485     A
BDT                LX2    -JT.SUCCP                                                     FCC0514     486     A
BDT                SA4    B5+X2        LI = TB + SUCC[T]                                FCC0514     487     A
BDT                LX4    -I.PRIP                                                       FCC0514     488     A
BDT                SB7    X4           FWA = PRI[LI] */ FWA OF EDGE GROUP               FCC0514     489     A
BDT                SX1    B6-B7                                                         BDT        1267     A
BDT                CALL   SHL          SORT( FWA , LWA-FWA )                            BDT        1268          I
          -CCG1
BDT                CALL   SHL=         SORT( FWA , LWA-FWA )                            CCG1        295          I
          -CCGNC
BDT                CALL   SHL#         SORT( FWA , LWA-FWA )                            CCGNC       805     A
BDT       RIO12    BSS    0                                                             BDT        1269          I
          -FCC0514
BDT                                                                                    FCC0514     490     A
BDT       *        IF CALL FROM *GPO*, THEN LINK ST PRED TO ST"S SO THAT *SIO*          FCC0514     491     A
BDT       *        CAN ARRANGE THE INSTRUCTIONS SO THAT THE ST,S FOLLOW THEIR RI.       FCC0514     492     A
BDT                                                                                    FCC0514     493     A
BDT       RIO12    SA2    O.TXT                                                         FCC0514     494     A
BDT                SA1    RMODE                                                         FCC0514     495     A
BDT                SB7    X2+3         TB = O.TXT + 3                                   FCC0514     496     A
BDT                ZR     X1,SIO       IF RMODE = 0                                     FCC0514     497     A
BDT                                                                                    FCC0514     498     A
BDT                SA3    L.TXT                                                         FCC0514     499     A
BDT                SB4    4                                                             FCC0514     500     A
BDT                SB3    X3-4         I = L.TXT - 4                                    FCC0514     501     A
BDT                MX1    -I.LVLL                                                       FCC0514     502     A
BDT                SA5    B7           LI = TB                                          FCC0514     503     A
BDT                LX1    I.LVLP                                                        FCC0514     504     A
BDT                SB6    B7-B1                                                         FCC0514     505     A
BDT                                                                                    FCC0514     506     A
BDT       *        CLEAR *LVL* FIELDS OF LINK WORDS                                     FCC0514     507     A
BDT                                                                                    FCC0514     508     A
BDT       RIO13    SA5    A5+B4        LI = LI + 4                                      FCC0514     509     A
BDT                SB3    B3-B4        I = I - 4                                        FCC0514     510     A
BDT                BX6    X1*X5                                                         FCC0514     511     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                 SA6    A5          LVL[LI] = 0                                  FCC0514      512     A
BDT                 NZ     B3,RIO13    IF I " 0                                     FCC0514      513     A
BDT                                                                                 FCC0514      514     A
BDT                 SA4    B7-B1                                                    FCC0514      515     A
BDT                 SX7    B1                                                       FCC0514      516     A
BDT                 LX7    D.BMP                                                    FCC0514      517     A
BDT                 BX6    X7+X4       BM[TXT+2] = 1                                FCC0514      518     A
BDT                 SA6    A4                                                       FCC0514      519     A
BDT                 MX0    -R1.RIL                                                  FCC0514      520     A
BDT                                                                                 FCC0514      521     A
BDT       *         SCAN PRED OF BM"S FOR ST"S, AND FORM CHAIN FROM ST PRED TO ST"   FCC0514      522     A
BDT                                                                                 FCC0514      523     A
BDT        RIO14    LX5    -I.NPP                                                   FCC0514      524     A
BDT                 SB5    X5          N = NP[LI]                                   FCC0514      525     A
BDT                 LX5    I.NPP-I.PRIP                                             FCC0514      526     A
BDT                 SA3    X5+         TI = PRI[LI];  TE = [TI]                     FCC0514      527     A
BDT                                                                                 FCC0514      528     A
BDT       *         SCAN EDGE LIST FOR PREVIOUS *BM*                                FCC0514      529     A
BDT                                                                                 FCC0514      530     A
BDT        RIO15    LX3    -JT.PREDP                                                FCC0514      531     A
BDT                 SA4    B6+X3       DI = TB-1 + PRED[TE]                         FCC0514      532     A
BDT                 SB3    X3          RMIN = PRED[TE]  */ TXT INDEX OF PREV *BM*   FCC0514      533     A
BDT                 SA3    A3+B1       TI = TI + 1;  TE = [TI]                      FCC0514      534     A
BDT                 LX4    59-D.BMP                                                 FCC0514      535     A
BDT                 PL     X4,RIO15    IF ^BM[DI]                                   FCC0514      536     A
BDT                                                                                 FCC0514      537     A
BDT                 SA3    X5-1        TI = PRI[LI] - 1                             FCC0514      538         I
          -FCC2273X
BDT                 SX7    B3                                                       FCC0514      539         I
          -FCC2273X
BDT                 SA7    RL          RL = RMIN                                    FCC0514      540         I
          -FCC2273X
BDT                 SA7    A7+B1       R(2) = RMIN                          002050  FCC2157        1         I
          -FCC2273X
BDT                 LX7    I.LVLP                                                   FCC0514      541         I
          -FCC2273X
BDT                                                                                 FCC0514      542         I
          -FCC2273X
BDT       *         CHECK FOR MULTIPLE DEFS, AND LINK THEM TO LAST DEF               FCC0514      543         I
          -FCC2273X
BDT                                                                                 FCC0514      544         I
          -FCC2273X
BDT        RIO16    LX4    D.BMP-D.^DP                                              FCC0514      545         I
          -FCC2273X
BDT                 MI     X4,RIO17    IF ^D[DI]     */ NOT A DEF                   FCC0514      546         I
          -FCC2273X
BDT                 SA4    A4-B4       DI = DI - 4                                  FCC0514      547         I
          -FCC2273X
BDT                 SA5    A4+B1       LI = DI + 1                                  FCC0514      548         I
          -FCC2273X
BDT                 SB3    B3-B4       RMIN = RMIN - 4                              FCC0514      549         I
          -FCC2273X

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                 LX4    59-D.BMP                                                  FCC0514      550          I
          -FCC2273X
BDT                 BX6    X7+X5       LVL[LI] = RL  */ LINK THIS DEF TO LAST DEF    FCC0514      551          I
          -FCC2273X
BDT                 SA6    A5                                                        FCC0514      552          I
          -FCC2273X
BDT                 MI     X4,RIO16    IF BM[DI]     */ NOT END OF *BM*"S            FCC0514      553          I
          -FCC2273X
BDT                                                                       002070    FCC2157        2          I
          -FCC2273X
BDT                 SX7    B3                                             002080    FCC2157        3          I
          -FCC2273X
BDT                 SA7    RL+1        RL(2)=RMIN                         002090    FCC2157        4          I
          -FCC2273X
BDT                 EQ     RIO17                                          002100    FCC2157        5          I
          -FCC2273X
BDT                                                                               FCC0514      554          I
          -FCC2273X
BDT       *         SCAN EDGE LIST OF *BM* FOR *ST*"S THAT ARE LINKED TO IT         FCC0514      555          I
          -FCC2273X
BDT                                                                               FCC0514      556          I
          -FCC2273X
BDT        RIO16A   SB3    X3          RMIN = PRED[TI]                    002120    FCC2157        6          I
          -FCC2273X
BDT                                                                       002130    FCC2157        7          I
          -FCC2273X
BDT        RIO17    ZR     B5,RIO19    IF N = 0      */ END OF EDGE LIST             FCC0514      557          I
          -FCC2273X
BDT                 SA3    A3+B1       TI = TI + 1;  TE = [TI]                         FCC0514      558          I
          -FCC2273X
BDT                 SB5    B5-B1       N = N - 1                                       FCC0514      559          I
          -FCC2273X
BDT                 LX3    -JT.PREDP                                                 FCC0514      560          I
          -FCC2273X
BDT                 SA4    B6+X3       DI = TB-1 + PRED[TE]                            FCC0514      561          I
          -FCC2273X
BDT                 LX4    59-D.STP                                                  FCC0514      562          I
          -FCC2157  -FCC2273X
BDT                 LX4    59-D.BMP                                           002150    FCC2157        8          I
          -FCC2273X
BDT                 MI     X4,RIO16A   IF BM[DI]                              002160    FCC2157        9          I
          -FCC2273X
BDT                 LX4    D.BMP-D.STP                                        002170    FCC2157       10          I
          -FCC2273X
BDT                 PL     X4,RIO17    IF ^ST[DI]    */ NOT A STORE                  FCC0514      563          I
          -FCC2273X
BDT                 IFNE   .DAL,0,2                                                   FCC0514      564          I
          -FCC2273X
BDT                 LX4    D.STP-D.L2P                                                FCC0514      565          I
          -FCC2273X
BDT                 MI     X4,RIO17    IF L2[DI]                                      FCC0514      566          I
          -FCC2273X

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                 SA4    A4-2        R1 = DI - 2                                    FCC0514      567          I
          -FCC2273X
BDT                 BX6    -X0*X4      R = RI[R1]                                     FCC0514      568          I
          -FCC2273X
BDT                 SB2    X6                                                         FCC0514      569          I
          -FCC2273X
BDT                 LT     B2,B3,RIO17 IF R < RMIN   */ PRED BEFORE PREV *BM*         FCC0514      570          I
          -FCC2273X
BDT                 SA3    X5-1        TI = PRI[LI] - 1                               FCC2273X       9     A
BDT                 SX7    B3                                                         FCC2273X      10     A
BDT                 SA7    RL          RL = RMIN                                      FCC2273X      11     A
BDT                 LX7    I.LVLP                                                     FCC2273X      12     A
BDT                                                                                   FCC2273X      13     A
BDT       *         CHECK FOR MULTIPLE DEFS, AND LINK THEM TO LAST DEF                FCC2273X      14     A
BDT                                                                                   FCC2273X      15     A
BDT        RIO16    LX4    D.BMP-D.^DP                                                FCC2273X      16     A
BDT                 MI     X4,RIO16A   IF ^D[DI] */NOT A DEF                          FCC2273X      17     A
BDT                 SA4    A4-B4       DI = DI - 4                                    FCC2273X      18     A
BDT                 SA5    A4+B1       LI = DI + 1                                    FCC2273X      19     A
BDT                 SB3    B3-B4       RMIN = RMIN - 4                                FCC2273X      20     A
BDT                 LX4    59-D.BMP                                                   FCC2273X      21     A
BDT                 BX6    X7+X5       LVL[LI] = RL  */ LINK THIS DEF TO LAST DEF     FCC2273X      22     A
BDT                 SA6    A5                                                         FCC2273X      23     A
BDT                 MI     X4,RIO16    IF BM[DI]     */ NOT END OF *BM*"S             FCC2273X      24     A
BDT                                                                                   FCC2273X      25     A
BDT        RIO16A   SX7    B3                                                         FCC2273X      26     A
BDT                 SA7    A7+B1       RL(2) = RMIN                                   FCC2273X      27     A
BDT                 EQ     RIO17                                                      FCC2273X      28     A
BDT                                                                                   FCC2273X      29     A
BDT       *         SCAN EDGE LIST OF *BM* FOR *ST*"S THAT ARE LINKED TO IT           FCC2273X      30     A
BDT                                                                                   FCC2273X      31     A
BDT        RIO16B   SB3    X3          RMIN = PRED[TI]                                FCC2273X      32     A
BDT                                                                                   FCC2273X      33     A
BDT        RIO17    ZR     B5,RIO19    IF N = 0      */ END OF EDGE LIST              FCC2273X      34     A
BDT                 SA3    A3+B1       TI = TI + 1;  TE = [TI]                        FCC2273X      35     A
BDT                 SB5    B5-B1       N = N - 1                                      FCC2273X      36     A
BDT                 LX3    -JT.PREDP                                                  FCC2273X      37     A
BDT                 SA4    B6+X3       DI = TB-1 + PRED[TE]                           FCC2273X      38     A
BDT                 LX4    59-D.BMP                                                   FCC2273X      39     A
BDT                 MI     X4,RIO16B   IF BM[DI]                                      FCC2273X      40     A
BDT                 LX4    D.BMP-D.STP                                                FCC2273X      41     A
BDT                 PL     X4,RIO17    IF ^ST[DI]    */ NOT A STORE                   FCC2273X      42     A
BDT                 IFNE   .DAL,0,2                                                   FCC2273X      43     A
BDT                 LX4    D.STP-D.L2P                                                FCC2273X      44     A
BDT                 MI     X4,RIO17    IF L2[DI]                                      FCC2273X      45     A
BDT                 SA4    A4-2        R1 = DI - 2                                    FCC2273X      46     A
BDT                 BX6    -X0*X4      R = RI[R1]                                     FCC2273X      47     A
BDT                 SB2    X6                                                         FCC2273X      48     A
BDT                 LT     B2,B3,RIO17 IF R< RMIN */PRED BEFORE PREV *BM*             FCC2273X      49     A
BDT                                                                                   FCC0514      571     A
BDT       *         DO NOT ADD ST TO CHAIN IF IT IS LINKED TO AN INTERFERING          FCC0514      572     A
BDT       *         LD OR ST BETWEEN IT AND ITS PRED ( RI ).                          FCC0514      573          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

          -CCG14
BDT       *         LD OR ST BETWEEN IT AND ITS PRED ( RI ), OR AN RJXJ .          CCG14        10     A
BDT                                                                              FCC0514     574     A
BDT                 SA5    A4+3        LI = R1 + 3                               FCC0514     575     A
BDT                 LX5    -I.NPP                                                FCC0514     576     A
BDT                 SB2    X5          NSP = NP[LI]                              FCC0514     577     A
BDT                 LX5    I.NPP-I.PRIP                                          FCC0514     578     A
BDT                 MX2    -D.LDSTL                                              FCC0514     579     A
BDT                 SA5    X5          TJ = PRI[LI]                              FCC0514     580     A
BDT                 LX2    D.LDSTP                                               FCC0514     581     A
BDT                                                                              FCC0514     582     A
BDT        RIO17A   LX5    -JT.PREDP   J = PRED[TJ]                              FCC0514     583     A
BDT                 SX5    X5                                                    FCC0514     584     A
BDT                 IX7    X6-X5                                                 FCC0514     585     A
BDT                 PL     X7,RIO17B   IF J @ R      */ PRED IS BEFORE RI        FCC0514     586     A
BDT                 SA4    B6+X5       DJ = TB-1 + J                             FCC0514     587     A
BDT                 BX7    -X2*X4                                                FCC0514     588     A
BDT                 LX4    59-D.ECJP                                             CCG14        11     A
BDT                 MI     X4,RIO17    IF ECJ[DJ]    */ EC JUMP IN BETWEEN       CCG14        12     A
BDT                 ZR     X7,RIO17B   IF ^LDST[DJ]  */ NOT A MEM REF            FCC0514     589     A
BDT                 LX4    59-D.EQVP                                             FCC0514     590         I
          -CCG14
BDT                 LX4    D.ECJP-D.EQVP                                         CCG14        13     A
BDT                 MI     X4,RIO17    IF EQV[DJ]    */ INTERFERING MEM REF      FCC0514     591     A
BDT        RIO17B   SB2    B2-B1       NSP = NSP - 1                             FCC0514     592     A
BDT                 SA5    A5+B1       TJ = TJ + 1                               FCC0514     593     A
BDT                 NZ     B2,RIO17A   IF NSP " 0                                FCC0514     594     A
BDT                                                                              FCC0613       6     A
BDT       *         DO NOT LINK *ST* TO PRED IF IT IS A *DEF*, AND RF IS AN EXPR  FCC0613       7     A
BDT       *         SINCE WE MAY GET IN TROUBLE WHEN THERE ARE REDEFINITION LINKS. FCC0613       8     A
BDT                                                                              FCC0613       9     A
BDT                 SA4    B6+X6       DIP = TB-1 + R                            FCC0613      10     A
BDT                 LX4    59-D.PIP                                              FCC0613      11     A
BDT                 PL     X4,RIO18    IF ^PI[DIP]   */ RI NOT A *DEF*           FCC0613      12     A
BDT                                                                              FCC0613      13     A
BDT                 SA4    B6+X3       DI = TB-1 + PRED[TI]                      FCC0613      14     A
BDT                 LX4    58-D.TYP                                              FCC0613      15     A
BDT                 SA2    A4-B1       R2 = DI - 1                               FCC0613      16     A
BDT                 LX2    -IH.RFP     RF = RF[R2]                               FCC0613      17     A
BDT                 MI     X4,RIO17C   IF TYPE[DI] = III                         FCC0613      18     A
BDT                 SA2    A2-B1       R1 = R2 - 1                               FCC0613      19     A
BDT                 LX2    -R1.RJP     RF = RJ[R1]                               FCC0613      20     A
BDT        RIO17C   BX2    -X0*X2                                                FCC0613      21     A
BDT                 SB2    X2                                                    FCC0613      22     A
BDT                 LE     B2,B1,RIO18 IF RF @ 1     */ NO RF                    FCC0613      23     A
BDT                 SA4    B6+B2       DIF = TB-1 + RF                           FCC0613      24     A
BDT                 LX4    59-D.ZPP                                              FCC0613      25     A
BDT                 PL     X4,RIO17    IF ^ZP[DIF]   */ NOT A SIMPLE *LD*        FCC0613      26     A
BDT                                                                              FCC0514     595     A
BDT       *         SCAN ST PRED CHAIN FOR END, AND ADD ST TO IT.                 FCC0514     596     A
BDT                                                                              FCC0514     597     A
BDT        RIO18    SA5    B7+X6       LI = TB + R                               FCC0514     598     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                BX6    -X1*X5      R = LVL[LI]                                  FCC0514      599     A
BDT                LX6    -I.LVLP                                                  FCC0514      600     A
BDT                NZ     X6,RIO18    IF R " 0      */ NOT END OF CHAIN            FCC0514      601     A
BDT                                                                                FCC0514      602     A
BDT                SX7    X3                                                       FCC0514      603     A
BDT                LX7    I.LVLP                                                   FCC0514      604     A
BDT                BX6    X7+X5       LVL[LI] = PRED[TE]                           FCC0514      605     A
BDT                SA6    A5                                                       FCC0514      606     A
BDT                EQ     RIO17                                                    FCC0514      607     A
BDT                                                                                FCC0514      608     A
BDT      *         CLEAR ST CHAIN PTRS FOR ALL BUT LAST DEF IN A SUB-SEQUENCE      FCC0514      609     A
BDT                                                                                FCC0514      610     A
BDT       RIO19    SA3    RL                                                       FCC0514      611     A
BDT                SB5    X3                                                       FCC0514      612           I
          -FCC2157
BDT                SA4    A3+B1                                          002190    FCC2157       11     A
BDT                SB5    X3                                             002200    FCC2157       12     A
BDT                SB3    X4          RMIN = RL(2)                       002210    FCC2157       13     A
BDT       RIO20    EQ     B3,B5,RIO21 IF RL = RMIN                                 FCC0514      613     A
BDT                SB5    B5-B4       RL = RL - 4                                  FCC0514      614     A
BDT                SA4    B7+B5       LI = TB + RL                                 FCC0514      615     A
BDT                BX7    X1*X4       LVL[LI] = 0   */ CLEAR *ST* CHAIN LINK       FCC0514      616     A
BDT                SA7    A4                                                       FCC0514      617     A
BDT                EQ     RIO20                                                    FCC0514      618     A
BDT                                                                                FCC0514      619     A
BDT      *         ADVANCE TO PREVIOUS *BM*                                        FCC0514      620     A
BDT                                                                                FCC0514      621     A
BDT       RIO21    SA5    B7+B3       LI = TB + RMIN                               FCC0514      622     A
BDT                NZ     B3,RIO14    IF RMIN " 0   */ NOT *BOS*                   FCC0514      623     A
BDT                                                                                FCC0514      624     A
BDT                SA4    B7-B1                                                    FCC0514      625     A
BDT                SX7    B1                                                       FCC0514      626     A
BDT                LX7    D.BMP                                                    FCC0514      627     A
BDT                BX6    -X7*X4      BM[TXT+2] = 0                                FCC0514      628     A
BDT                SA6    A4                                                       FCC0514      629     A
BDT                LX7    D.PSP-D.BMP                                              FCC0514      630     A
BDT                SX6    B1                                                       FCC0514      631     A
BDT                LX6    D.PIP                                                    FCC0514      632     A
BDT                BX7    X6+X7                                                    FCC0514      633     A
BDT                SA7    RMODE       RMODE = (PS,PI)                              FCC0514      634     A
BDT       SIO      EJECT              SIO                                          BDT         1270     A
BDT      *         SIO - SET INSTRUCTION ORDER BY A DEPTH FIRST SEARCH             BDT         1271     A
BDT                                                                                BDT         1272     A
BDT                SA2    O.TXT                                                    BDT         1273           I
          -FCC0514
BDT                SA3    L.TXT                                                    BDT         1274           I
          -FCC0514
BDT                SB7    X2+3        TB = TXT + 3                                 BDT         1275           I
          -FCC0514
BDT       SIO      SA3    L.TXT                                                    FCC0514      635     A
BDT                SB4    60-I.PRIP                                                BDT         1276     A
BDT                SX7    X3-4        N = L.TXT - 4 */ CURRENT INSTRUCTION         BDT         1277     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                SB5    4           ON = 4        */ OUTPUT INDEX                    BDT         1278     A
BDT                SA4    RMODE                                                        BDT         1279     A
BDT                MX0    1                                                            BDT         1280     A
BDT                LX0    1+I.MIP                                                      BDT         1281     A
BDT                SX1    B1                                                           BDT         1282     A
BDT                SB6    0           LII = 0       */ LAST ISSUED INSTRUCTION         BDT         1283     A
BDT                SKIP   1                                                  ****      BDT         1284          I
          -FCC0514
BDT                ZR     X4,SIO2     IF RMODE = 0                                     BDT         1285          I
          -FCC0514
BDT                NZ     X4,SIO2     IF RMODE " 0  */ CALL FROM *GPO/CBB*             FCC0514      636     A
BDT                                                                                    BDT         1286     A
BDT       *        SEARCH FOR NEXT TERMINAL INSTRUCTION                                BDT         1287     A
BDT                                                                                    BDT         1288          I
          -FCC0514
BDT                                                                                    BDT         1289     A
BDT        SIO1    SA2    RIOA                                                         BDT         1290     A
BDT                SB3    4                                                            BDT         1291     A
BDT                SB2    B7-B1       DB = TB - 1                                      BDT         1292     A
BDT                SA4    B2+B6       DI = DB + LII                                    BDT         1293     A
BDT                                                                                    BDT         1294     A
BDT        SIO1A   SA4    A4+B3       DI = DI + 4                                      BDT         1295     A
BDT                BX6    X2*X4                                                        BDT         1296     A
BDT                ZR     X6,SIO1A    IF ^( BM[DI] ! ST[DI] )                          BDT         1297     A
BDT                                                                                    BDT         1298     A
BDT                SB6    A4-B2       LII = DI - DB                                    BDT         1299     A
BDT                SX7    A4-B2       N = LII                                          BDT         1300     A
BDT                                                                                    BDT         1301     A
BDT       *        VISIT CURRENT INSTRUCTION                                           BDT         1302     A
BDT                                                                                    BDT         1303     A
BDT        SIO2    SA5    B7+X7       LI = TB + N                                      BDT         1304     A
BDT                LX4    B4,X5                                                        BDT         1305          I
          -FCC0514
BDT                SA2    RMODE                                                        FCC0514      637     A
BDT                SA4    A5-B1       DI = LI - 1                                      FCC0514      638     A
BDT                BX6    X2*X4                                                        FCC0514      639     A
BDT                ZR     X6,SIO2D    IF ^RMODE[DI] */ NOT *PS* RO *PI*                FCC0514      640          I
          -FCC1126
BDT                MX2    -I.LVLL                                                      FCC0514      641          I
          -FCC1126
BDT                LX2    I.LVLP                                                       FCC0514      642          I
          -FCC1126
BDT                BX6    -X2*X5                                                       FCC0514      643          I
          -FCC1126
BDT                ZR     X6,SIO2D    IF LVL[LI] = 0  */ ST"S PROCESSED                FCC0514      644          I
          -FCC1126
BDT                ZR     X6,SIO2D    IF ^RMODE[DI] */ NOT *PS* ! *PI*                 FCC1126       24     A
BDT                BX6    X5                                                           FCC1126       25     A
BDT                LX6    59-I.STCPP                                                   FCC1126       26     A
BDT                MI     X6,SIO2D    IF STCP[LI]   */ STC PROCESSED                   FCC1126       27     A
BDT                LX6    1+I.STCPP-I.LVLP                                             FCC1126       28     A
BDT                SX2    X6                                                           FCC1126       29     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                 ZR     X2,SIO2D    IF LVL[LI] = 0  */ NO ST CHAIN (EOQ/EOS)      FCC1126      30     A
BDT                                                                                  FCC0514     645     A
BDT       *         ADD ST"S TO THE STACK SO WE OUTPUT THEM IMMEDIATELY AFTER         FCC0514     646     A
BDT       *         THEIR PREDECESSORS ( RI AND RF )                                  FCC0514     647     A
BDT                                                                                  FCC0514     648     A
BDT                 SX1    X7          T = N         */ SAVE N                        FCC0514     649         I
          -FCC1126
BDT                 LX6    -I.LVLP                                                   FCC0514     650         I
          -FCC1126
BDT                 SX7    X6          N = LVL[LI]                                   FCC0514     651         I
          -FCC1126
BDT                 BX6    X2*X5                                                     FCC0514     652         I
          -FCC1126
BDT                 SA6    A5          LVL[LI] = 0                                   FCC0514     653         I
          -FCC1126
BDT                 LX1    I.STCPP                                                   FCC1126      31     A
BDT                 BX6    X1+X5       STCP[LI] = 1                                  FCC1126      32     A
BDT                 SX1    X7          T = N         */ SAVE N                       FCC1126      33     A
BDT                 LX6    -I.LVLP                                                   FCC1126      34     A
BDT                 SX7    X6          N = LVL[LI]                                   FCC1126      35     A
BDT                 BX6    X6-X7                                                     FCC1126      36     A
BDT                 LX6    I.LVLP      LVL[LI] = 0                                   FCC1126      37     A
BDT                 SA6    A5                                                        FCC1126      38     A
BDT                                                                                  FCC0514     654     A
BDT       SIO2A     SA5    B7+X7       LI = TB + N                                   FCC0514     655     A
BDT                 BX6    X0*X5                                                     FCC0514     656     A
BDT                 NZ     X6,SIO2C    IF MI[LI]     */ ST PROCESSED                 FCC0514     657     A
BDT                                                                                  FCC0514     658     A
BDT                 SA7    A0+B1       [SI+1] = N    */ STACK N                      FCC0514     659     A
BDT                 SA0    A0+B1       SI = SI + 1                                   FCC0514     660     A
BDT                                                                                  FCC0514     661     A
BDT                 SA4    A5-B1       DI = LI - 1                                   FCC0514     662     A
BDT                 SA3    A4-B1       R2 = DI -1                                    FCC0514     663     A
BDT                 LX4    58-D.TYP                                                  FCC0514     664     A
BDT                 LX3    -IH.RFP     R = RF[R2]                                    FCC0514     665     A
BDT                 MI     X4,SIO2AA   IF TYPE[DI] = III                             FCC0514     666     A
BDT                 SA3    A3-B1       R1 = R2 - 1                                   FCC0514     667     A
BDT                 MX4    -R1.RIL                                                   FCC0514     668     A
BDT                 LX3    -R1.RJP                                                   FCC0514     669     A
BDT                 BX3    -X4*X3      R = RJ[R1]                                    FCC0514     670     A
BDT       SIO2AA    SB2    X3                                                        FCC0514     671     A
BDT                 LE     B2,B1,SIO2C IF R @ 1      */ NO RF                        FCC0514     672     A
BDT                                                                                  FCC0514     673     A
BDT                 SA4    B7+B2       LIP = TB + R                                  FCC0514     674     A
BDT                 LX4    59-I.MIP                                                  FCC0514     675     A
BDT                 PL     X4,SIO2C    IF ^MI[LIP]                                   FCC0514     676         I
          -FCC1908
BDT                 LX4    I.MIP-I.IIP                                               FCC0514     677         I
          -FCC1908
BDT                 MI     X4,SIO2A1   IF M[LIP]                                     FCC1908       5     A
BDT                 SX4    B2                                                        FCC1908       6     A
BDT                 IX4    X4-X1                                                     FCC1908       7     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                MI     X4,SIO2C    IF T > R                                  FCC1908        8     A
BDT                                                                             FCC1908        9     A
BDT                SA0    A0-B1       SI = SI - 1   */ REMOVE N FROM STACK      FCC1908       10     A
BDT                PRINT  SIO,(* REMOVE STORE STACK --*)                        FCC1908       11     A
BDT                EQ     SIO2C                                                 FCC1908       12     A
BDT                                                                             FCC1908       13     A
BDT       SIO2A1   LX4    I.MIP-I.IIP                                           FCC1908       14     A
BDT                MI     X4,SIO2C    IF II[LIP]    */ ISSUED                   FCC0514      678     A
BDT                                                                             FCC0514      679     A
BDT      *         RF MARKED BUT NOT OUTPUT, ADD *N* BELOW IT IN THE STACK      FCC0514      680     A
BDT                                                                             FCC0514      681     A
BDT                SA4    A0-B1       SJ = SI - 1                               FCC0514      682     A
BDT                SB2    -B2                                                   FCC0514      683     A
BDT       SIO2B    BX6    X4          S = [SJ]                                  FCC0514      684     A
BDT                SX3    B2+X4                                                 FCC0514      685     A
BDT                SA6    A4+B1       [SJ+1] = S                                FCC0514      686     A
BDT                SA4    A4-B1       SJ = SJ - 1                               FCC0514      687     A
BDT                NZ     X3,SIO2B    IF S " R                                  FCC0514      688     A
BDT                SA7    A4+B1       [SJ+1] = N    */ INSERT N BELOW R         FCC0514      689     A
BDT                                                                             FCC0514      690     A
BDT       SIO2C    LX5    -I.LVLP                                               FCC0514      691     A
BDT                SX7    X5          N = LVL[LI]                               FCC0514      692     A
BDT                NZ     X7,SIO2A    IF N " 0      */ NOT END OF ST CHAIN      FCC0514      693     A
BDT                                                                             FCC0514      694     A
BDT                SX7    X1          N = T */ RESTORE N                        FCC0514      695     A
BDT                SX1    B1                                                    FCC0514      696     A
BDT                SA5    B7+X7       LI = TB + N                               FCC0514      697     A
BDT                                                                             FCC0514      698     A
BDT       SIO2D    LX4    B4,X5                                                 FCC0514      699     A
BDT                BX5    X0+X5       MI[LI] = 1                                BDT         1306     A
BDT                LX3    B1,X5                                                 BDT         1307     A
BDT                ERRNZ  58-I.IIP                                              BDT         1308     A
BDT                SB3    X4-1        EI = PRI[LI] - 1  */ EDGE INDEX           BDT         1309     A
BDT                MI     X3,SIO5     IF II[LI]     */ ISSUED                   BDT         1310     A
BDT                                                                             BDT         1311     A
BDT      *         SEARCH PRED TO SEE IF ALL ARE MARKED                         BDT         1312     A
BDT                                                                             BDT         1313     A
BDT       SIO3     SA2    B3+X5       TI = EI + NP[LI]                          BDT         1314     A
BDT                SB2    X5                                                    BDT         1315     A
BDT                ZR     B2,SIO4     IF NP[LI] = 0                             BDT         1316     A
BDT                                                                             BDT         1317     A
BDT                IX5    X5-X1       NP[LI] = NP[LI] - 1                       BDT         1318     A
BDT                LX2    -JT.PREDP   R = PRED[TI]                              BDT         1319     A
BDT                SA3    B7+X2       LIP = TB + R                              BDT         1320     A
BDT                LX3    59-I.MIP                                              BDT         1321          I
          -FCC0514
BDT                MI     X3,SIO3     IF MI[LIP]    */ VISITED OR OUTPUT        BDT         1322          I
          -FCC0514
BDT                LX4    B1,X3                                                 FCC0514      700     A
BDT                SX2    X2                                                    FCC0514      701     A
BDT                MI     X4,SIO3     IF II[LIP]    */ ISSUED                   FCC0514      702     A
BDT                LX3    59-I.MIP                                              FCC0514      703     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                MI     X3,SIO3A    IF MI[LIP]                                    FCC0514      704     A
BDT                IX6    X2-X7                                                     FCC0514      705     A
BDT                PL     X6,SIO3A    IF R > N      */ REDEF LINK                   FCC0514      706     A
BDT                                                                                BDT         1323     A
BDT                SA7    A0+B1       [SI+1] = N                                    BDT         1324          I
         -FCC1126
BDT       SIO3H    SA7    A0+B1       [SI+1] = N    */ STACK CURRENT INST           FCC1126       39     A
BDT                SA0    A0+B1       SI = SI + 1                                   BDT         1325     A
BDT                BX6    X5                                                        BDT         1326     A
BDT                SA6    A5                                                        BDT         1327     A
BDT                SX7    X2          N = R         */ FOLLOW PRED                  BDT         1328     A
BDT                EQ     SIO2                                                      BDT         1329     A
BDT                                                                                FCC1126       40     A
BDT       SIO3G    IX6    X2-X7                                                     FCC1126       41     A
BDT                PL     X6,SIO3     IF R > N      */ REDEF LINK                   FCC1126       42     A
BDT                PRINT  SIO,(* STACK N --- MARKED INST ---*)                      FCC1126       43     A
BDT                EQ     SIO3H                                                     FCC1126       44     A
BDT                                                                                BDT         1330          I
         -FCC0514
BDT                                                                                FCC0514      707     A
BDT      *         *R* NOT OUTPUT, IF TOP OF STACK IS A REDEF *ST*, THEN SET *XU*  FCC0514      708     A
BDT      *         BIT IN IT, OR MOVE IT DOWN IN THE STACK AFTER R, SO WE AVOID    FCC0514      709     A
BDT      *         THE PROBLEM OF USES EXTENDING PAST A DEFINITION IN *GRA*.       FCC0514      710     A
BDT      *         SEE SECTION ON DEADLOCKS IN THE *IMS* FOR FURTHER INFORMATION.  FCC0514      711     A
BDT                                                                                FCC0514      712     A
BDT       SIO3A    SA3    A0          SJ = SI;  S = [SJ]                            FCC0514      713     A
BDT                SB2    B7-B1                                                     FCC0514      714          I
         -FCC0791
BDT                BX4    X5                                                        FCC0791        7     A
BDT                SB2    B7-B1                                                     FCC0791        8     A
BDT                LX4    59-I.RDSTP                                                FCC0791        9     A
BDT                MI     X4,SIO3AA   IF RDST[LI]   */ REDEF ST IN THE STACK        FCC0791       10          I
         -FCC1126
BDT                SA4    B2+X3       DIS = TB-1 + S                                FCC0514      715          I
         -FCC1126
BDT                LX4    59-D.STP                                                  FCC0514      716          I
         -FCC1126
BDT                PL     X4,SIO3     IF ^ST[DIS]     */ NOT A STORE                FCC0514      717          I
         -FCC1126
BDT                SA4    A4-2        R1S = DIS - 2                                 FCC0514      718          I
         -FCC1126
BDT                MX6    -R1.RIL                                                   FCC0514      719          I
         -FCC1126
BDT                BX6    -X6*X4                                                    FCC0514      720          I
         -FCC1126
BDT                BX4    X7-X6                                                     FCC0514      721          I
         -FCC1126
BDT                NZ     X4,SIO3     IF RI[R1S] " N  */ NEXT INST ^ *ST  N,, *     FCC0514      722          I
         -FCC1126
BDT                                                                                FCC0514      723          I
         -FCC1126
BDT                SA4    B7+X2       LIP = TB + R                                  FCC0514      724          I

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
          -FCC0791  -FCC1126
BDT                 SX4    B1                                                  FCC0791      11          I
          -FCC1126
BDT                 LX4    I.RDSTP                                             FCC0791      12          I
          -FCC1126
BDT                 BX5    X4+X5       RDST[LI] = 1  */ INDICATE REDEF ST IN STK  FCC0791      13          I
          -FCC1126
BDT                                                                            FCC0791      14          I
          -FCC1126
BDT        SIO3AA   SA4    B7+X2       LIP = TB + R                            FCC0791      15          I
          -FCC1126
BDT                 LX4    59-I.MIP                                            FCC0514     725          I
          -FCC1126
BDT                 PL     X4,SIO3D    IF ^MI[LIP]   */ R NOT IN THE STACK     FCC0514     726          I
          -FCC1126
BDT                                                                            FCC0514     727          I
          -FCC1126
BDT        SIO3B    SA3    A3-B1       SJ = SJ - 1;  S = [SJ]                  FCC0514     728          I
          -FCC1126
BDT                 SA4    B2+X3       DJ = TB-1 + S                           FCC0514     729          I
          -FCC1126
BDT                 BX6    X3-X2                                               FCC0514     730          I
          -FCC1126
BDT                 LX4    59-D.BMP                                            FCC0514     731          I
          -FCC1126
BDT                 MI     X4,SIO3D    IF BM[DJ]       */ DONT MOVE ST PAST A *BM*  FCC0514     732          I
          -FCC1126
BDT                 NZ     X6,SIO3B    IF S " R        */ NOT END OF SEARCH    FCC0514     733          I
          -FCC1126
BDT                                                                            FCC0514     734          I
          -FCC1126
BDT       *         MOVE TOP OF STACK UP TO DELAY OUTPUT OF N+4 UNTIL AFTER *R*  FCC0514     735          I
          -FCC1126
BDT                                                                            FCC0514     736          I
          -FCC1126
BDT                 SA3    A0+         SJ = SI                                 FCC0514     737          I
          -FCC1126
BDT                 SB2    X3+         SAVE = [SI]                             FCC0514     738          I
          -FCC1126
BDT                                                                            FCC0514     739          I
          -FCC1126
BDT        SIO3C    SA3    A3-B1       SJ = SJ - 1;  S = [SJ]                  FCC0514     740          I
          -FCC1126
BDT                 BX4    X3-X2                                               FCC0514     741          I
          -FCC1126
BDT                 SX6    X3                                                  FCC0514     742          I
          -FCC1126
BDT                 SA6    A3+B1       [SJ+1] = S                              FCC0514     743          I
          -FCC1126
BDT                 NZ     X4,SIO3C    IF S " R                                FCC0514     744          I
          -FCC1126
BDT                                                                            FCC0514     745          I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
          -FCC1126
BDT                 SX6    B2                                                    FCC0514     746          I
          -FCC1126
BDT                 SA6    A3          [SJ] = SAVE     */ RESTORE *ST* IN STACK  FCC0514     747          I
          -FCC1126
BDT                 EQ     SIO3                                                  FCC0514     748          I
          -FCC1126
BDT                                                                              FCC0514     749          I
          -FCC1126
BDT       *         REDEF CANT BE MOVED BACK, SET *XU* BIT IN DESCR FOR *GRA*    FCC0514     750          I
          -FCC1126
BDT                                                                              FCC0514     751          I
          -FCC1126
BDT        SIO3D    SA3    A0                                                    FCC0514     752          I
          -FCC0791  -FCC1126
BDT                 SA4    B2+X3       DIS = TB-1 + [SI]                         FCC0514     753          I
          -FCC0791  -FCC1126
BDT        SIO3D    SA3    A0+B1       SJ = SI + 1                               FCC0791      16          I
          -FCC1126
BDT                 MX6    -R1.RIL                                               FCC0791      17          I
          -FCC1126
BDT        SIO3E    SA3    A3-B1      SJ = SJ - 1                                FCC0791      18          I
          -FCC1126
BDT                 SA4    B2+X3       DIS = TB-1 + [SJ]                         FCC0791      19          I
          -FCC1126
BDT                 LX4    59-D.STP                                              FCC0791      20          I
          -FCC1126
BDT                 PL     X4,SIO3E    IF ^ST[DIS]                               FCC0791      21          I
          -FCC1126
BDT                 SA4    A4-2        R1S = DIS - 2                             FCC0791      22          I
          -FCC1126
BDT                 BX3   X4-X7                                                  FCC0791      23          I
          -FCC1126
BDT                 BX4    -X6*X3                                                FCC0791      24          I
          -FCC1126
BDT                 NZ     X4,SIO3E    IF RI[R1S] " N  */ NOT * ST  N,, *        FCC0791      25          I
          -FCC1126
BDT                                                                              FCC0791      26          I
          -FCC1126
BDT                 SA4    A4+2        DIS = R1S + 2                             FCC0791      27          I
          -FCC1126
BDT                 MI     X4,SIO3AA   IF RDST[LI]   */ NOT FIRST REDEF EDGE     FCC1126      45          I
          -FCC1908
BDT                 MI     X4,SIO3B    IF RDST[LI]   */ NOT FIRST REDEF EDGE     FCC1908      15     A
BDT                 SA4    B2+X3       DIS = TB-1 + S                            FCC1126      46     A
BDT                 LX4    59-D.RDP                                              FCC1126      47     A
BDT                 PL     X4,SIO3G    IF ^RD[DIS]   */ NOT A REDEF STORE        FCC1126      48     A
BDT                 SA4    A4-2                                                  FCC1126      49     A
BDT                 MX6    -R1.RIL                                               FCC1126      50     A
BDT                 BX6    -X6*X4                                                FCC1126      51     A
BDT                 BX4    X7-X6                                                 FCC1126      52     A
BDT                 NZ     X4,SIO3G    IF RI[R1S] " N  */ TOP OF STK ^ *ST N,,*  FCC1126      53     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                                                                                  FCC1126      54     A
BDT                 SX4    B1                                                        FCC1126      55     A
BDT                 LX4    I.RDSTP                                                   FCC1126      56     A
BDT                 BX5    X4+X5       RDST[LI] = 1  */ INDICATE REDEF ST ST IN STK  FCC1126      57     A
BDT                 LX3    I.LVLP                                                    FCC1126      58     A
BDT                 BX5    X3+X5       LVL[LI] = S                                   FCC1126      59     A
BDT                 LX3    -I.LVLP                                                   FCC1126      60     A
BDT                                                                                  FCC1126      61     A
BDT       SIO3AA    SA4    B7+X2       LIP = TB + R                                  FCC1126      62           I
         -FCC1908
BDT                 LX4    59-I.MIP                                                  FCC1126      63           I
         -FCC1908
BDT                 PL     X4,SIO3F    IF ^MI[LIP]   */ R NOT IN THE STACK           FCC1126      64           I
         -FCC1908
BDT                                                                                  FCC1126      65           I
         -FCC1908
BDT      *          LOOK AT THE STACK AND SEE IF WE CAN MOVE THE STORE BELOW R       FCC1126      66           I
         -FCC1908
BDT      *          SO IT COMES OUT AFTER IT.                                        FCC1126      67           I
         -FCC1908
BDT                                                                                  FCC1126      68           I
         -FCC1908
BDT                 SA4    A0          SJ = SI       */ SEARCH FOR STORE IN STK      FCC1126      69           I
         -FCC1908
BDT                 LX5    -I.LVLP                                                   FCC1126      70           I
         -FCC1908
BDT                 SX3    X5+         S = LVL[LI]   */ TXT INDEX OF *ST*            FCC1126      71           I
         -FCC1908
BDT                 LX5    I.LVLP                                                    FCC1126      72           I
         -FCC1908
BDT       SIO3B     BX6    X3-X4                                                     FCC1126      73           I
         -FCC1908
BDT                 SA4    A4-B1       SJ = SJ - 1                                   FCC1126      74           I
         -FCC1908
BDT                 NZ     X6,SIO3B    IF [SJ+1] " S                                 FCC1126      75           I
         -FCC1908
BDT                                                                                  FCC1126      76           I
         -FCC1908
BDT                 SX4    A4+B1       SS = SJ + 1                                   FCC1126      77           I
         -FCC1908
BDT                 SA3    A0+         SJ = SI       */ SEARCH FOR *R*               FCC1126      78           I
         -FCC1908
BDT       SIO3C     BX6    X2-X3                                                     FCC1126      79           I
         -FCC1908
BDT                 SA3    A3-B1       SJ = SJ - 1                                   FCC1126      80           I
         -FCC1908
BDT                 NZ     X6,SIO3C    IF [SJ+1] " R                                 FCC1126      81           I
         -FCC1908
BDT                 SX3    A3+B1                                                     FCC1126      82           I
         -FCC1908
BDT                 IX6    X4-X3                                                     FCC1126      83           I
         -FCC1908

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                 MI     X6,SIO3     IF SS < SJ+1  */ S IS ALREADY BELOW R          FCC1126      84          I
          -FCC1908
BDT                 PRINT  SIO,(21X*MOVE ST DOWN*2Z6),(X4,X3)                         FCC1126      85          I
          -FCC1908
BDT                                                                                   FCC1126      86          I
          -FCC1908
BDT                 SA3    X4          SJ = SS       */ MOVE THE STORE DOWN           FCC1126      87          I
          -FCC1908
BDT                 SX1    X3          S = [SJ]                                        FCC1126      88          I
          -FCC1908
BDT        SIO3D    SA3    A3-B1       SJ = SJ - 1                                     FCC1126      89          I
          -FCC1908
BDT                 SA4    B2+X3       DIS = TB-1 + [SJ]                               FCC1126      90          I
          -FCC1908
BDT                 BX6    X3                                                          FCC1126      91          I
          -FCC1908
BDT                 SA6    A3+B1       [SJ+1] = [SJ]                                   FCC1126      92          I
          -FCC1908
BDT                 LX4    59-D.BMP                                                    FCC1126      93          I
          -FCC1908
BDT                 MI     X4,SIO3E    IF BM[DIS]    */ CANT MOVE ST PAST A *BM*       FCC1126      94          I
          -FCC1908
BDT                 IX4    X2-X3                                                       FCC1126      95          I
          -FCC1908
BDT                 NZ     X4,SIO3D    IF [SJ] " R   */ IF NOT FINISHED                FCC1126      96          I
          -FCC1908
BDT                                                                                   FCC1126      97          I
          -FCC1908
BDT                 SX6    X1                                                          FCC1126      98          I
          -FCC1908
BDT                 SA6    A3          [SJ] = S                                        FCC1126      99          I
          -FCC1908
BDT                 SX1    B1                                                          FCC1126     100          I
          -FCC1908
BDT                 EQ     SIO3                                                        FCC1126     101          I
          -FCC1908
BDT                                                                                   FCC1126     102          I
          -FCC1908
BDT        SIO3E    SX6    X1                                                          FCC1126     103          I
          -FCC1908
BDT                 SA6    A6          [SJ+1] = S                                      FCC1126     104          I
          -FCC1908
BDT                 SX1    1                                                           FCC1126     105          I
          -FCC1908
BDT                                                                                   FCC1126     106          I
          -FCC1908
BDT       *         REDEF CANNOT BE MOVED BACK, SET *XU* BIT IN DESCR FOR *GRA*       FCC1126     107          I
          -FCC1908
BDT                                                                                   FCC1126     108          I
          -FCC1908
BDT        SIO3F    BX6    X5                                                          FCC1126     109          I
          -FCC1908

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                 LX6    -I.LVLP                                                  FCC1126      110          I
          -FCC1908
BDT                 SA4    B2+X6       DIS = TB-1 + LVL[LI]   */ DESCR OF *ST*      FCC1126      111          I
          -FCC1908
BDT                 PRINT  SIO,(21X*SET XU*Z5),X6                                   FCC1126      112          I
          -FCC1908
BDT                 SX3    B1                                                       FCC0514      754          I
          -FCC1908
BDT                 LX3    D.XUP                                                    FCC0514      755          I
          -FCC1908
BDT                 BX6    X3+X4       XU[DIS] = 1                                  FCC0514      756          I
          -FCC1908
BDT                 SA6    A4                                                       FCC0514      757          I
          -FCC1908
BDT                 EQ     SIO3                                                     FCC0514      758          I
          -FCC1908
BDT        SIO3B    RJ     MSD         MOVE THE STORE DOWN / SET THE *XU* BIT       FCC1908       16     A
BDT                 EQ     SIO3                                                     FCC1908       17     A
BDT                                                                                 FCC0514      759     A
BDT       *         OUTPUT INSTRUCTION                                              BDT         1331     A
BDT                                                                                 BDT         1332     A
BDT        SIO4     LX1    I.IIP       II[LI] = 1                                   BDT         1333     A
BDT                 SA4    A5-B1       DI = LI - 1                                  BDT         1334     A
BDT                 SX2    B5          OI[LI] = ON                                  BDT         1335     A
BDT                 LX2    I.OIP                                                    BDT         1336     A
BDT                 BX3    X1+X2                                                    BDT         1337     A
BDT                 SX1    B1                                                       BDT         1338     A
BDT                 BX6    X3+X5                                                    BDT         1339     A
BDT                 LX4    59-D.PRSP                                                BDT         1340     A
BDT                 SA6    A5                                                       BDT         1341     A
BDT                 SB5    B5+4        ON = ON + 4                                  BDT         1342     A
BDT                 PL     X4,SIO5     IF ^PRS[DI]                                  BDT         1343          I
          -FCC0514
BDT        SIO      SPACE  1,10                                                     BDT         1344          I
          -FCC0514
BDT       *         INSTRUCTION *PRS*, OUTPUT THE *RS*                              BDT         1345          I
          -FCC0514
BDT                                                                                 BDT         1346          I
          -FCC0514
BDT                 SA5    A5+4        LI = LI + 4                                  BDT         1347          I
          -FCC0514
BDT                 MI     X4,SIO4A    IF PRS[DI]                                   FCC0514      760     A
BDT                 LX4    D.PRSP-D.RJRSP                                           FCC0514      761     A
BDT                 PL     X4,SIO5     IF ^RJRS[DI]                                 FCC0514      762          I
          -FCC0642
BDT                                                                                 FCC0514      763          I
          -FCC0642
BDT       *         INSTRUCTION *PRS* OR *RJRS*, OUTPUT THE *RS*                    FCC0514      764          I
          -FCC0642
BDT                 MI     X4,SIO4A    IF RJRS[DI]                                  FCC0642        5     A
BDT                 LX4    D.RJRSP-D.XMTP                                           FCC0642        6     A
BDT                 PL     X4,SIO5     IF ^XMT[DI]                                  FCC0642        7     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

BDT                 SA4    A4+4        DI = DI + 4   */ CHECK NEXT FOR *KS*          FCC0642        8     A
BDT                 LX4    59-D.KSP                                                  FCC0642        9     A
BDT                 PL     X4,SIO5     IF ^KS[DI]                                    FCC0642       10     A
BDT                                                                                  FCC0642       11     A
BDT       *         INSTRUCTION *PRS*, *RJRS* OR IS A *SXT*, OUTPUT THE NEXT INST    FCC0642       12     A
BDT                                                                                  FCC0514      765     A
BDT        SIO4A    SA5    A5+4        LI = LI + 4                                   FCC0514      766     A
BDT                 MX2    -I.OIL                                                    BDT         1348     A
BDT                 LX2    I.OIP                                                     BDT         1349     A
BDT                 BX6    X2*X5                                                     BDT         1350     A
BDT                 BX5    X0+X6       MI[LI] = 1                                    BDT         1351     A
BDT                 EQ     SIO4                                                      BDT         1352     A
BDT                                                                                  BDT         1353     A
BDT        SIO5     SA3    A0          N = [ST]                                      BDT         1354     A
BDT                 SA0    A0-1        ST = ST - 1                                   BDT         1355     A
BDT                 SX7    X3                                                        BDT         1356     A
BDT                 NZ     X3,SIO2     IF N " 0                                      BDT         1357     A
BDT                                                                                  BDT         1358     A
BDT                 SA4    A5-3        R1 = LI - 3                                   BDT         1359     A
BDT                 SA0    A0+B1       ST = ST + 1                                   BDT         1360     A
BDT                 UX6    B2,X4                                                     BDT         1361     A
BDT                 NZ     B2,SIO1     IF OC[R1] " OC.EOQ                            BDT         1362     A
BDT                                                                                  BDT         1363     A
BDT                 TRACE  SIO,RLIST                                                 BDT         1364     A
BDT                                                                                  BDT         1365         I
          -FCC0514
BDT        SIO      EJECT                                                            FCC0514      767         I
          -FCC0791
BDT        SIO      SPACE  2,10                                                      FCC0791       28     A
BDT       *         MOVE THE INSTRUCTIONS TO THEIR FINAL ORDER                       BDT         1366     A
BDT                                                                                  BDT         1367     A
BDT                 SA1    L.TXT                                                     BDT         1368     A
BDT                 SA3    O.TREE                                                    BDT         1369     A
BDT                 SB6    B7-B1                    (B6) = TXT + 2                   BDT         1370     A
BDT                 MOVE   X1-8,B7+B1,X3     MOVE( L.TXT-8 , TXT+4 , TREE )          BDT         1371     A
BDT                 SA1    N.INST                                                    BDT         1372     A
BDT                 SA2    O.TXT                                                     BDT         1373     A
BDT                 SA3    O.TREE                                                    BDT         1374     A
BDT                 SB3    3                                                         BDT         1375     A
BDT                 SA5    X3+B3       TI = TREE + 3                                 BDT         1376     A
BDT                 SB7    X3                       (B7) = TREE                      BDT         1377     A
BDT                 SX1    X1-2        N = N.INST - 2                                BDT         1378     A
BDT                 SX0    B1                                                        BDT         1379     A
BDT                                                                                  BDT         1380     A
BDT        SIO8     SA4    A5-B1       D = [TI-1]                                    BDT         1381     A
BDT                 SA3    A4-B1       R2 = [TI-2]                                   BDT         1382     A
BDT                 SA2    A5-B3       R1 = [TI-3]                                   BDT         1383     A
BDT                 LX5    -I.OIP                                                    BDT         1384     A
BDT                 SB5    X5          J = OI[TI]                                    BDT         1385         I
          -CCG0215
BDT                 MI     X5,SIO9                  */ IF MARKED                     CCG0215        7     A
BDT                 BX5    X2                         */ IF NOT MARKED, USE RI       CCG0215        8     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                                                                                    CCG0215        9     A
BDT       SIO9     SB5    X5          J = OI[TI]                                       CCG0215       10     A
BDT                SA5    A5+4        TI = TI + 4                                      BDT         1386     A
BDT                BX6    X4                                                           BDT         1387     A
BDT                LX7    X3                                                           BDT         1388     A
BDT                SA6    B6+B5       TXT(J+2) = D                                     BDT         1389     A
BDT                IX1    X1-X0       N = N - 1                                        BDT         1390     A
BDT                SA7    A6-B1       TXT(J+1) = R2                                    BDT         1391     A
BDT                BX6    X2                                                           BDT         1392     A
BDT                SA6    A7-B1       TXT(J) = R1                                      BDT         1393     A
BDT                NZ     X1,SIO8     IF N " 0                                         BDT         1394     A
BDT                                                                                    BDT         1395     A
BDT                MX7    0                                                            BDT         1396     A
BDT                SA7    L.TREE      L.TREE = 0                                       BDT         1397     A
BDT                SA0    B6-2                                                         BDT         1398     A
BDT                SX2    0                                                            BDT         1399     A
BDT                CALL   RNI         RENUMBER THE INSTRUCTIONS                        BDT         1400     A
BDT                SA5    O.TXT                                                        BDT         1401          I
          -FCC0514
BDT                                                                                    FCC0514      768     A
BDT       SIO10    SA5    O.TXT                                                        FCC0514      769     A
BDT                RJ     CRW         COMPUTE X-REGISTER WIDTH OF NEW ORDER            BDT         1402     A
BDT                EQ     RIO                                                          BDT         1403     A
BDT                                                                                    BDT         1404     A
BDT       RIOA     BFMW   D,(BM,ST)                                                    BDT         1405     A
BDT       RMODE    BSS    1           *RIO* MODE                                       BDT         1406     A
BDT       RL       BSS    1           *TXT* INDEX OF LAST *BM*                         FCC0514      770          I
          -FCC2157
BDT       RL        BSS    2                                                  002230   FCC2157       14     A
BDT       MSD      SPACE  3                                                            FCC1908       18     A
BDT      **        MSD - MOVE STORE ( TOP OF STACK ) DOWN TO A POINT BELOW *R*.        FCC1908       19     A
BDT                                                                                    FCC1908       20     A
BDT       MSD      ROUTINE                                                             FCC1908       21     A
BDT                SA4    B7+X2       LIP = TB + R                                     FCC1908       22     A
BDT                LX4    59-I.MIP                                                     FCC1908       23     A
BDT                PL     X4,MSD5     IF ^MI[LIP]   */ R NOT IN THE STACK              FCC1908       24     A
BDT                                                                                    FCC1908       25     A
BDT      *         SEE IF WE CAN MOVE *TOS* BELOW *R* SO IT COMES OUT AFTER IT         FCC1908       26     A
BDT                                                                                    FCC1908       27     A
BDT                SA4    A0          SJ = SI       */ SEARCH FOR STORE IN STK         FCC1908       28     A
BDT                LX5    -I.LVLP                                                      FCC1908       29     A
BDT                SX3    X5          S = LVL[LI]   */ TXT INDEX OF *ST*               FCC1908       30     A
BDT                LX5    I.LVLP                                                       FCC1908       31     A
BDT       MSD1     BX6    X3-X4                                                        FCC1908       32     A
BDT                SA4    A4-B1       SJ = SJ - 1                                      FCC1908       33     A
BDT                NZ     X6,MSD1     IF [SJ+1] " S                                    FCC1908       34     A
BDT                                                                                    FCC1908       35     A
BDT                SX4    A4+B1       SS = SJ + 1                                      FCC1908       36     A
BDT                SA3    A0+         SJ = SI       */ SEARCH FOR *R*                  FCC1908       37     A
BDT       MSD2     BX6    X2-X3                                                        FCC1908       38     A
BDT                SA3    A3-B1       SJ = SJ - 1                                      FCC1908       39     A
BDT                NZ     X6,MSD2     IF [SJ+1] " R                                    FCC1908       40     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                SX3    A3+B1                                                FCC1908      41     A
BDT                IX6    X4-X3                                                FCC1908      42     A
BDT                MI     X6,MSD      IF SS < SJ+1  */ S IS ALREADY BELOW R    FCC1908      43     A
BDT                PRINT  SIO,(21X*MOVE STORE DOWN*)                           FCC1908      44     A
BDT                                                                            FCC1908      45     A
BDT                SA3    X4          SJ = SS       */ MOVE THE STORE DOWN    FCC1908      46     A
BDT                SX1    X3          S = [SJ]                                 FCC1908      47     A
BDT        MSD3    SA3    A3-B1       SJ = SJ - 1                              FCC1908      48     A
BDT                SA4    B2+X3       DIS = TB-1 + [SJ]                        FCC1908      49     A
BDT                BX6    X3                                                   FCC1908      50     A
BDT                SA6    A3+B1       [SJ+1] = [SJ]                            FCC1908      51     A
BDT                LX4    59-D.BMP                                             FCC1908      52     A
BDT                MI     X4,MSD4     IF BM[DIS]    */ CANT MOVE ST PAST A *BM* FCC1908      53     A
BDT                IX4    X2-X3                                                FCC1908      54     A
BDT                NZ     X4,MSD3     IF [SJ] " R   */ IF NOT FINISHED         FCC1908      55     A
BDT                                                                            FCC1908      56     A
BDT                SX6    X1                                                   FCC1908      57     A
BDT                SA6    A3          [SJ] = S                                 FCC1908      58     A
BDT                SX1    B1                                                   FCC1908      59     A
BDT                EQ     MSD                                                  FCC1908      60     A
BDT                                                                            FCC1908      61     A
BDT        MSD4    SX6    X1                                                   FCC1908      62     A
BDT                SA6    A6          [SJ+1] = S                               FCC1908      63     A
BDT                SX1    1                                                    FCC1908      64     A
BDT                                                                            FCC1908      65     A
BDT        *       REDEF CANNOT BE MOVED BACK, SET *XU* BIT IN DESCR FOR *GRA* FCC1908      66     A
BDT                                                                            FCC1908      67     A
BDT        MSD5    BX6    X5                                                   FCC1908      68     A
BDT                LX6    -I.LVLP                                              FCC1908      69     A
BDT                SA4    B2+X6       DIS = TB-1 + LVL[LI]  */ DESCR OF STORE  FCC1908      70     A
BDT                PRINT  SIO,(21X*SET XU*Z5),X6                               FCC1908      71     A
BDT                SX3    B1                                                   FCC1908      72     A
BDT                LX3    D.XUP                                                FCC1908      73     A
BDT                BX6    X3+X4       XU[DIS] = 1                              FCC1908      74     A
BDT                SA6    A4                                                   FCC1908      75     A
BDT                EQ     MSD                                                  FCC1908      76     A
BDT         CRW    TITLE  CRW - COMPUTE REGISTER WIDTH OF A SEQUENCE           BDT        1407     A
BDT        **      CRW - COMPUTE THE X-REGISTER WIDTH OF A INSTRUCTION SEQUENCE BDT        1408     A
BDT        *       AND INSTALL IT IN THE DESCRIPTORS.                          BDT        1409     A
BDT        *                                                                   BDT        1410     A
BDT        *       ENTRY  (X5) = FWA OF RLIST INSTRUCTION BLOCK                BDT        1411     A
BDT        *                                                                   BDT        1412     A
BDT        *       EXIT   (X6) = MAX REG WIDTH OF SEQUENCE                     BDT        1413     A
BDT                                                                            BDT        1414     A
BDT         CRW    ENTRY. **                                                   BDT        1415          I
           -CCGNC
BDT         CRW    ENTRY. **,#                                                 CCGNC       806     A
BDT                MX0    -R1.RIL+1                                            BDT        1416          I
           -CCG11
BDT                SB4    B0          RW = 0        */ REG WIDTH               BDT        1417     A
BDT                MX1    -D.REGWL                                             BDT        1418     A
BDT                LX0    1                                                    BDT        1419          I

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

          -CCG11
     BDT                SX0    RN.MASK                                                  CCG11        136     A
     BDT                SB5    B0          MW = 0        */ MAX WIDTH                   BDT         1420     A
     BDT                SB6    B0          LXW = 0       */ LOCKED X-REG WIDTH     000610  CCG32          6     A
     BDT                LX1    D.REGWP                                                  BDT         1421     A
     BDT                MX2    -D.USESL                                                 BDT         1422     A
     BDT                SB7    X5+3        TB = TXT + 3                                 BDT         1423     A
     BDT                SA4    B7-B1       DI = TB - 1                                  BDT         1424     A
     BDT                SB3    OC.RS                                                    BDT         1425     A
     BDT                MX7    0                                                        BDT         1426     A
     BDT                SA7    B7          [TB] = 0                                     BDT         1427     A
     BDT                EQ     CRW1                                                     BDT         1428     A
     BDT                                                                               BDT         1429     A
     BDT       *        ^D OR PRS TO A B-REG, INSTALL REGW, CLEAR LINK WORD             BDT         1430     A
     BDT                                                                               BDT         1431     A
     BDT        CRW0    LX4    1+D.^DP                                                  BDT         1432     A
     BDT                SX3    B4                                                       BDT         1433     A
     BDT                BX4    X1*X4                                                    BDT         1434     A
     BDT                LX3    D.REGWP                                                  BDT         1435     A
     BDT                MX7    0                                                        BDT         1436     A
     BDT                BX6    X3+X4       REGW[DI] = RW                                BDT         1437     A
     BDT                SA6    A4                                                       BDT         1438     A
     BDT                SA7    A4+B1       [DI+1] = 0                                   BDT         1439     A
     BDT                                                                               BDT         1440     A
     BDT       *        ADVANCE TO NEXT INSTRUCTION, JUMP ON TYPE                       BDT         1441     A
     BDT                                                                               BDT         1442     A
     BDT        CRW1    SA5    A4+2        R1 = DI + 2                                  BDT         1443     A
     BDT                SA4    A5+2        DI = R1 + 2                                  BDT         1444     A
     BDT                UX5    B2,X5                                                    BDT         1445     A
     BDT                LX4    58-D.TYP                                                 BDT         1446     A
     BDT                IX7    X4+X4                                                    BDT         1447     A
     BDT                MI     X4,CRW3     IF TYPE[DI] = III ! IV                       BDT         1448     A
     BDT                MI     X7,CRW2     IF TYPE[DI] = II                             BDT         1449     A
     BDT                                                                               BDT         1450     A
     BDT       *        TYPE I                                                          BDT         1451     A
     BDT                                                                               BDT         1452     A
     BDT                LX7    1+D.TYP-D.TYIOP                                          BDT         1453     A
     BDT                MX3    -D.TYIOL                                                 BDT         1454     A
     BDT                LX4    1+D.TYP-D.^DP                                            BDT         1455     A
     BDT                LT     B2,B3,CRW1A IF OC[R1] < OC.RS  */ *DAR*                  BDT         1456     A
     BDT                BX6    -X3*X7                                                   BDT         1457     A
     BDT                SA3    =XTYIOM+X6                                               BDT         1458          I
          -CCGNC
     BDT                SA3    =XTYIOM#+X6                                              CCGNC        807     A
     BDT                UX7    B2,X3                                                    BDT         1459     A
     BDT                BX6    X7*X5                                                    BDT         1460     A
     BDT                AX5    B2,X6                                                    BDT         1461     A
     BDT                AX5    B2,X6       RNUMS = TYIOM[DI,R1]                         BDT         1462     A
     BDT                EQ     CRW7                                                     BDT         1463     A
     BDT                                                                               BDT         1464     A
     BDT       *        PROCESS *DAR*                                                   BDT         1465     A
     BDT                                                                               BDT         1466     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT       CRW1A    AX5    R1.RJP                                                BDT        1467    A
BDT                BX7    -X0*X5      R = RJ[R1]                                BDT        1468    A
BDT                SA3    B7+X7       LI = TB + R                               BDT        1469    A
BDT                SX6    X3-1        [LI] = [LI] - 1                           BDT        1470    A
BDT                SA6    A3                                                    BDT        1471    A
BDT                NZ     X6,CRW0     IF [LI] " 0   */ NOT LAST USE             BDT        1472    A
BDT                SB4    B4-B1       RW = RW - 1                               BDT        1473    A
BDT                EQ     CRW0                                                  BDT        1474    A
BDT                                                                             BDT        1475    A
BDT      *         TYPE II                                                       BDT        1476    A
BDT                                                                             BDT        1477    A
BDT       CRW2     BX6    X4                                                    BDT        1478    A
BDT                LX4    1+D.TYP-D.^DP                                         BDT        1479    A
BDT                LX6    1+D.TYP-D.PIP                                         BDT        1480    A
BDT                PL     X6,CRW9     IF ^PI[DI]                                BDT        1481    A
BDT                                                                             BDT        1482    A
BDT                EQ     B2,B3,CRW0  IF OC[R1] = OC.RS                         BDT        1483    A
BDT                LX5    58-R1.SOP-SO.RTP                                      BDT        1484    A
BDT                PL     X5,CRW0     IF SORT[R1] " 2  */ DEF NOT TO AN X-REG   BDT        1485    A
BDT                                                                             BDT        1486    A
BDT                LX5    SO.RTP-SO.LKP                                 000630  CCG32         7    A
BDT                PL     X5,CRW9     IF SOLK[R1] < 1 */ NOT FULL OR A-LOCK     000640  CCG32         8    A
BDT                SB6    B6+1        LXW = LXW + 1                             000650  CCG32         9    A
BDT                EQ     CRW9                                                  BDT        1487    A
BDT                                                                             BDT        1488    A
BDT      *         TYPE IV                                                       BDT        1489    A
BDT                                                                             BDT        1490    A
BDT       CRW3     PL     X7,CRW4     IF TYPE[DI] " IV                          BDT        1491    A
BDT                ZR     B2,CRW11    IF OC[R1] = OC.EOQ                        BDT        1492    A
BDT                                                                             BDT        1493    A
BDT                LX4    1+D.TYP-D.^DP                                         BDT        1494    A
BDT                SX6    B2-OC.EOS                                             BDT        1495         I
          -FCC0883
BDT                ZR     X6,CRW0     IF OC[R1] = OC.EOS                        BDT        1496         I
          -FCC0883
BDT                SX6    B2-OC.ENT                                             FCC0883       4    A
BDT                MI     X6,CRW0     IF OC[R1] < OC.ENT  */ EOS OR LAB         FCC0883       5    A
BDT                SB4    0           RW = 0        */ CLEAR UJP LOCKS          BDT        1497         I
          -CCG32
BDT                SB4    B6          RW = LXW      */ RESET, CLEAR UJP LOCKS   000670  CCG32        10    A
BDT                EQ     CRW0                                                  BDT        1498    A
BDT                                                                             BDT        1499    A
BDT      *         TYPE III                                                      BDT        1500    A
BDT                                                                             BDT        1501    A
BDT       CRW4     LX4    1+D.TYP-D.^DP     RNUMS = RI[R1]                      BDT        1502    A
BDT                MI     X4,CRW5     IF ^D[DI]                                 BDT        1503    A
BDT                MX5    0           RNUMS = 0                                 BDT        1504    A
BDT       CRW5     SA3    A5+B1       R2 = R1 + 1                               BDT        1505    A
BDT                BX5    -X0*X5                                                BDT        1506    A
BDT                LX3    -IH.RFP                                               BDT        1507    A
BDT                BX6    -X0*X3      R = RF[R2]                                BDT        1508    A
BDT                ZR     X6,CRW7     IF R = 0                                  BDT        1509    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                LX5    R1.RIL                                          BDT        1510     A
BDT                BX5    X6+X5       RNUMS = SHIFT(RNUMS,R1.RIL) ! R     BDT        1511     A
BDT                                                                       BDT        1512     A
BDT       *        DECREMENT USES OF OPERANDS ( RNUMS )                   BDT        1513     A
BDT                                                                       BDT        1514     A
BDT        CRW7    BX7    -X0*X5      R = RI[RNUMS]                       BDT        1515     A
BDT                SA3    B7+X5       LI = TB + R                         BDT        1516     A
BDT                SX6    X3-1        [LI] = [LI] - 1                     BDT        1517     A
BDT                AX5    R1.RIL      RNUMS = SHIFT(RNUMS,-R1.RIL)        BDT        1518     A
BDT                NZ     X6,CRW8     IF [LI] " 0   */ NOT LAST USE       BDT        1519     A
BDT                SB4    B4-B1       RW = RW - 1                         BDT        1520     A
BDT        CRW8    SA6    A3                                              BDT        1521     A
BDT                NZ     X5,CRW7     IF RNUMS " 0                        BDT        1522     A
BDT        RIO     SPACE  1,10                                            BDT        1523     A
BDT       *        ADVANCE *RW* TO ACCOUNT FOR THE NEW RESULT             BDT        1524     A
BDT                                                                       BDT        1525     A
BDT        CRW9    MI     X4,CRW0     IF ^D[DI]                           BDT        1526     A
BDT                BX6    X4                                              BDT        1527     A
BDT                LX6    D.^DP-D.PRSP                                    BDT        1528     A
BDT                PL     X6,CRW10    IF ^PRS[DI]                         BDT        1529     A
BDT                SA3    A4+2        R1RS = DI + 2                       BDT        1530     A
BDT                LX3    58-R1.SOP-SO.RTP                                BDT        1531     A
BDT                PL     X3,CRW0     IF SORT[R1RS] " 2  */ NOT TO AN X-REG BDT      1532     A
BDT                                                                       BDT        1533     A
BDT        CRW10   LX4    1+D.^DP                                         BDT        1534     A
BDT                SB4    B4+B1       RW = RW + 1                         BDT        1535     A
BDT                BX4    X1*X4                                           BDT        1536     A
BDT                SX3    B4                                              BDT        1537     A
BDT                BX7    -X2*X4                                          BDT        1538     A
BDT                SA7    A4+B1       [DI+1] = USES[DI]                   BDT        1539     A
BDT                LX3    D.REGWP                                         BDT        1540     A
BDT                BX6    X3+X4       REGW[DI] = RW                       BDT        1541     A
BDT                SA6    A4                                              BDT        1542     A
BDT                LE     B4,B5,CRW1  IF RW @ MW                          BDT        1543     A
BDT                                                                       BDT        1544     A
BDT                SB5    B4          MW = RW                             BDT        1545     A
BDT                EQ     CRW1                                            BDT        1546     A
BDT        CRW     SPACE  1,10                                            BDT        1547     A
BDT       *        EOQ - SET MAX WIDTH IN DESCR OF *BOS* AND CLEAR LINK WORDS BDT    1548     A
BDT                                                                       BDT        1549     A
BDT        CRW11   SA3    B7-B1       DI = TB - 1                         BDT        1550     A
BDT                SX6    B5                                              BDT        1551     A
BDT                LX6    D.REGWP                                         BDT        1552     A
BDT                BX3    X1*X3                                           BDT        1553     A
BDT                MX7    0                                               BDT        1554     A
BDT                BX6    X3+X6       REGW[TB-1] = MW  */ SET MAXW IN *BOS* BDT      1555     A
BDT                SA6    A3                                              BDT        1556     A
BDT                                                                       BDT        1557     A
BDT                SA7    A3+B1                                           BDT        1558     A
BDT                SB2    A3+B1                                           BDT        1559     A
BDT                SB3    A4+B1                                           BDT        1560     A
BDT                SB4    4                                               BDT        1561     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT      +        SB2    B2+B4       FOR I = 3 TO L.TXT BY 4;  TXT(I) = 0          BDT      1562    A
BDT               SA7    A7+B4                                                     BDT      1563    A
BDT               LT     B2,B3,*                                                   BDT      1564    A
BDT               TRACE  CRW,RLIST                                                 BDT      1565    A
BDT                                                                                BDT      1566    A
BDT               SX6    B5          RIO.MAX = MW                                  BDT      1567    A
BDT               EQ     CRW                                                       BDT      1568    A
BDT       RNI     TITLE  RNI - RENUMBER INSTRUCTIONS                               BDT      1569    A
BDT      **       RNI - RENUMBER INSTRUCTIONS                                      BDT      1570    A
BDT      *        RELABEL THE R-NUMBERS IN THE INSTRUCTIONS SO THAT FOR EACH       BDT      1571    A
BDT      *        *RI* WHICH DEFINES A RESULT, RI = R1 - O.TXT ( THE INDEX TO      BDT      1572    A
BDT      *        *TXT* OF THE INSTRUCTION ). WITH THIS LABELING, R-NUMBERS        BDT      1573    A
BDT      *        APPEARING AS OPERANDS ARE POINTERS TO THE INSTRUCTIONS WHICH     BDT      1574    A
BDT      *        ARE THE OPERANDS.                                                BDT      1575    A
BDT      *                                                                         BDT      1576    A
BDT      *        ENTRY  (B7) = FWA OF SCRATCH TABLE USED FOR RENUMBERING          BDT      1577    A
BDT      *               (A0) = (O.TXT)                                            BDT      1578    A
BDT      *               (X2) = M = NEW BASE R-NUMBER                              BDT      1579    A
BDT      *                                                                         BDT      1580    A
BDT      *        EXIT   A0, B7 PRESERVED                                          BDT      1581    A
BDT      *               (B2) = 2                                                  BDT      1582    A
BDT      *               (B4) = 4                                                  BDT      1583    A
BDT      *               (X0) = M(-R1.RIL)                                         BDT      1584    A
BDT                                                                                BDT      1585    A
BDT       RNI     ENTRY.                                                           BDT      1586          I
         -CCGNC
BDT       RNI     ENTRY. **,#                                                      CCGNC     808    A
BDT               TRACE  RNI,RLIST                                                 BDT      1587    A
BDT               SB2    B1+B1       (B2) = 2                                      BDT      1588    A
BDT               MX0    -R1.RIL                                                   BDT      1589    A
BDT               SB5    59-D.^DP                                                  BDT      1590    A
BDT               SB4    B2+B2       (B4) = 4                                      BDT      1591    A
BDT               SA4    A0+B2       DI = O.TXT + 2                                BDT      1592    A
BDT               SX1    B4                                                        BDT      1593    A
BDT               BX7    X2                                                        BDT      1594    A
BDT               PX5    B2,X2                                                     BDT      1595    A
BDT                                                                                BDT      1596    A
BDT      *        SETUP THE EQUIVALENCE TABLE BETWEEN THE R-NUMBERS                BDT      1597    A
BDT                                                                                BDT      1598    A
BDT       RNI1    UX6    "OC",X5                                                  BDT      1599    A
BDT               ZR     "OC",RNI3   IF OC[R1] = 0 */ END OF *TXT*                BDT      1600    A
BDT                                                                                BDT      1601    A
BDT       RNI2    SA4    A4+B4       DI = DI + 4                                   BDT      1602    A
BDT               IX2    X2+X1       M = M + 4                                     BDT      1603    A
BDT               SA5    A4-B2       R1 = DI - 2                                   BDT      1604    A
BDT               LX6    B5,X4                                                     BDT      1605    A
BDT               MI     X6,RNI1     IF ^D[DI]                                     BDT      1606    A
BDT               BX3    -X0*X5      RI = RI[R1]                                   BDT      1607    A
BDT               IX6    X2-X3                                                     BDT      1608    A
BDT               LX7    X2          MAXR = M                                      BDT      1609    A
BDT               SA6    B7+X3       T(RI) = M - RI                                BDT      1610    A
BDT               EQ     RNI2                                                      BDT      1611    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT       RNI      SPACE  3                                                    BDT        1612     A
BDT      *         RELABEL THE R-NUMBERS IN THE INSTRUCTIONS USING THE EQV TABLE  BDT   1613     A
BDT                                                                            BDT        1614     A
BDT       RNI3     MX6    0                                                    BDT        1615     A
BDT                SA4    A0+B2       DI = O.TXT + 2                           BDT        1616     A
BDT                SB6    R1.RIL                                               BDT        1617     A
BDT                SA6    B7          T(0) = 0                                 BDT        1618     A
BDT                SA6    B7+B1       T(1) = 0                                 BDT        1619     A
BDT                EQ     RNI5                                                 BDT        1620     A
BDT       RNI      SPACE  1,10                                                 BDT        1621     A
BDT      *         PROCESS TYPE II / III INSTRUCTIONS                           BDT        1622     A
BDT                                                                            BDT        1623     A
BDT       RNI4     SA2    A5+B1       R2 = R1 + 1                              BDT        1624     A
BDT                BX7    -X0*X5                                               BDT        1625     A
BDT                SA3    B7+X7       RI = T(RI[R1])                           BDT        1626     A
BDT                IX6    X5+X3                                                BDT        1627     A
BDT                LX2    -IH.RFP                                              BDT        1628     A
BDT                SA3    B7+X2                                                BDT        1629     A
BDT                IX7    X2+X3       RF = T(RF[R2])                           BDT        1630     A
BDT                SA6    A5                                                   BDT        1631     A
BDT                LX7    IH.RFP                                               BDT        1632     A
BDT                SA7    A2+                                                  BDT        1633     A
BDT                                                                            BDT        1634     A
BDT       RNI5     SA4    A4+B4       DI = DI + 4                              BDT        1635     A
BDT                LX4    58-D.TYP                                             BDT        1636     A
BDT                SA5    A4-B2       RI = DI - 2                              BDT        1637     A
BDT                IX6    X4+X4                                                BDT        1638     A
BDT                BX7    X4-X6                                                BDT        1639     A
BDT                MI     X7,RNI4     IF TYPE[DI] = II ! III                   BDT        1640     A
BDT                MI     X4,RNI6     IF TYPE[DI] = IV                         BDT        1641     A
BDT                                                                            BDT        1642     A
BDT      *         PROCESS TYPE I INSTRUCTIONS                                  BDT        1643     A
BDT                                                                            BDT        1644     A
BDT                LX5    -R1.RJP                                              BDT        1645     A
BDT                BX1    -X0*X5                                               BDT        1646     A
BDT                SA2    B7+X1                                                BDT        1647     A
BDT                IX3    X5+X2       RJ = T( RJ[R1] )                         BDT        1648     A
BDT                LX4    B6,X3                                                BDT        1649     A
BDT                BX5    -X0*X4                                               BDT        1650     A
BDT                SA1    B7+X5                                                BDT        1651     A
BDT                IX7    X1+X4       RK = T( RK[R1] )                         BDT        1652     A
BDT                SA2    A5+B1       R2 = R1 + 1                              BDT        1653     A
BDT                LX6    B6,X7                                                BDT        1654     A
BDT                BX3    -X0*X6                                               BDT        1655     A
BDT                SA1    B7+X3                                                BDT        1656     A
BDT                IX7    X1+X6       RI = T( RI[R1] )                         BDT        1657     A
BDT                SA7    A5                                                   BDT        1658     A
BDT                ZR     X2,RNI5     IF [R2] = 0                              BDT        1659     A
BDT                LX2    -IH.RFP                                              BDT        1660     A
BDT                SA3    B7+X2                                                BDT        1661          I
         -FCC0643
BDT                BX7    -X0*X2                                               FCC0643      81     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

```
BDT                SA3    B7+X7                                                    FCC0643      82     A
BDT                LX7    X3                                                       CCG0177      11     A
BDT                AX7    IH.RFL                                                   CCG0177      12     A
BDT                NZ     X7,RNI5                 */ IF NOTHING SETUP IN EQV TABLE CCG0177      13     A
BDT                IX6    X2+X3       RF = T(RF[R2])                               BDT        1662     A
BDT                LX6    IH.RFP                                                   BDT        1663     A
BDT                SA6    A2                                                       BDT        1664     A
BDT                EQ     RNI5                                                     BDT        1665     A
BDT                                                                               BDT        1666     A
BDT       RNI6     UX7    "OC",X5                                                  BDT        1667     A
BDT                NZ     "OC",RNI5   IF OC[R1] " 0                                BDT        1668     A
BDT                EQ     RNI                                                      BDT        1669     A
BDT                                                                               BDT        1670     A
BDT                END                                                             BDT        1671          I
          -CCG7
BDT                ENTRY  F.BRDGE     FWA OF BRIDGE CODE ( OVERLAID )              CCG7        132          I
          -CCG11
BDT       F.BRDGE  END                                                             CCG7        133          I
          -CCG11
BDT                ENTRY  F$BRDGE     FWA OF BRIDGE CODE ( OVERLAID )              CCG11       137     A
BDT       F$BRDGE  END                                                             CCG11       138     A
```

SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - BDT

IDENTIFIER	TOTAL	ACTIVE
BDT	1671	1511
FCC0448	30	28
FCC0514	470	339
FCC0613	21	21
FCC0625	4	0
FCC0641	18	18
FCC0642	8	8
FCC0643	2	2
FCC0658	5	0
FCC0675	2	1
FCC0791	24	5
FCC0883	2	2
FCC0715A	8	8
FCC0881	20	10
FCC0977	62	61
FEAT155F	5	3
FCC1056	18	11
FCC1126	100	48
FCC1290	24	22
FCC1322	6	2
FCC1518X	20	20
CCG1	10	3
CCG3	29	29
CCG7	2	0
FCC1695X	6	6
CCGNC	15	14

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN BDT

                CCG11                        8         8
                FCC1739X                     2         2
                CCG14                        4         4
                FCC1825X                     4         4
                CCG18                        1         1
                FCC1908                     72        72
                CCG28                        2         2
                CCG32                        5         5
                FCC2157                     14         4
                CCGA010                      7         7
                CCGA017                      6         6
                FCC2273X                    41        41
                FCCA550X                     1         1
                CCG0177                      3         3
                CCG0188                      1         1
                CCG0207                      9         9
                CCG0215                      4         4

                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CCGDBG

      CCGDBG    *DECK     CCGDBG                                                  CCGDBG         1     A
      CCGDBG    *IF       DEF,TESTMODE                                            CCGDBG         2           I
                -CCGA054
      CCGDBG    *IF DEF,TEST                                                      CCGA054        9     A

           SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - CCGDBG

                IDENTIFIER                TOTAL    ACTIVE

                CCGDBG                       2         1
                CCGA054                      1         1

                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DMPIIT

      DMPIIT    *DECK     DMPIIT                                                  DMPIIT         1     A
      DMPIIT              SUBROUTINE DMPIIT( IIT , N , PIC )                      DMPIIT         2     A
      DMPIIT                                                                      DMPIIT         3     A
      DMPIIT    **        DMP IIT - FORMAT AND LIST INC VALUES AND INC FORMULAS IN *IIT*  DMPIIT  4     A
      DMPIIT                                                                      DMPIIT         5     A
      DMPIIT              IMPLICIT INTEGER(A-Z)                                   DMPIIT         6     A
      DMPIIT              COMMON/LINE/LINECM(2)                                   DMPIIT         7     A
      DMPIIT              INTEGER IIT(N)                                          DMPIIT         8     A
      DMPIIT              COMMON/BVINFO/VL                                        DMPIIT         9     A
      DMPIIT              DATA VECNAM / 5L  LE   /                                DMPIIT        10     A
      DMPIIT                                                                      DMPIIT        11     A
      DMPIIT              FIELD(X,P,L) = SHIFT(X,60-P) .A. .N.MASK(60-L)          DMPIIT        12     A
      DMPIIT    *                                INC VALUES                       DMPIIT        13     A
      DMPIIT              TYP(X) = FIELD(X,48,3)                                  DMPIIT        14     A
      DMPIIT              VUDI(X) = FIELD(X,36,12)                                DMPIIT        15     A
      DMPIIT              VCA(X) = FIELD(X,18,18)                                 DMPIIT        16     A
      DMPIIT              VIH(X) = FIELD(X,0,18)                                  DMPIIT        17     A
      DMPIIT    *                                IA WORD                          DMPIIT        18     A

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DMPIIT

```
DMPIIT              NWD(X) = FIELD(X,48,10)                                  DMPIIT      19    A
DMPIIT              CA(X) = FIELD(X,18,18)                                   DMPIIT      20    A
DMPIIT              UDI(X) = FIELD(X,1,17)                                   DMPIIT      21    A
DMPIIT    *                                IB WORD                           DMPIIT      22    A
DMPIIT              NLD(X) = FIELD(X,48,10)                                  DMPIIT      23    A
DMPIIT              ILD(X) = FIELD(X,0,18)                                   DMPIIT      24    A
DMPIIT    *                                     E. WORD OF *RXI*            DMPIIT      25    A
DMPIIT              SCR(X) = FIELD(X,59,1)                                   DMPIIT      26    A
DMPIIT              JP(X) = FIELD(X,58,1)                                    DMPIIT      27    A
DMPIIT              ALLP(X) = FIELD(X,57,1)                                  DMPIIT      28    A
DMPIIT              PRED(X) = FIELD(X,33,15)                                 DMPIIT      29    A
DMPIIT              BI(X) = FIELD(X,0,15)                                    DMPIIT      30    A
DMPIIT                                                                       DMPIIT      31    A
DMPIIT                                                                       DMPIIT      32    A
DMPIIT              PRINT 1                                                  DMPIIT      33    A
DMPIIT     1        FORMAT(/*   TYP  VAL, INCREMENT VALUES*/)               DMPIIT      34    A
DMPIIT              LINECM = LINECM(2)                                       DMPIIT      35    A
DMPIIT                                                                       DMPIIT      36    A
DMPIIT              DO 10 I = 1,PIC                                          DMPIIT      37    A
DMPIIT              T = TYP(IIT(I))                                          DMPIIT      38    A
DMPIIT              V = VIH(IIT(I))                                          DMPIIT      39    A
DMPIIT              PRINT 2,T                                                DMPIIT      40    A
DMPIIT     2        FORMAT(I5)                                               DMPIIT      41    A
DMPIIT              IF( T .NE. 0 ) GO TO 4                                   DMPIIT      42    A
DMPIIT              PRINT 3,V                                                DMPIIT      43    A
DMPIIT     3        FORMAT(Z8,*B*/)                                          DMPIIT      44    A
DMPIIT              GO TO 10                                                 DMPIIT      45    A
DMPIIT                                                                       DMPIIT      46    A
DMPIIT     4        ITW = IIT(I)                                             DMPIIT      47    A
DMPIIT              PRINT 5,VCA(ITW),SYM(V),VUDI(ITW)/2                      DMPIIT      48    A
DMPIIT     5        FORMAT(Z8,* + *A7,I5,*UDI*/)                             DMPIIT      49    A
DMPIIT     10       CONTINUE                                                 DMPIIT      50    A
DMPIIT                                                                       DMPIIT      51    A
DMPIIT    *         PRINT OUT THE INCREMENT FORMULAS                         DMPIIT      52    A
DMPIIT                                                                       DMPIIT      53    A
DMPIIT              I = PIC + 1                                              DMPIIT      54    A
DMPIIT     15       PRINT 11                                                 DMPIIT      55    A
DMPIIT     11       FORMAT(/)                                                DMPIIT      56    A
DMPIIT              X = IIT(I)                                               DMPIIT      57    A
DMPIIT              I = I + 1                                                DMPIIT      58    A
DMPIIT              IPASS = 0                                                DMPIIT      59    A
DMPIIT              PRINT 16,I-2,FIELD(X,59,1),FIELD(X,36,18),FIELD(X,18,18) ,  DMPIIT  60    A
DMPIIT         1    FIELD(X,0,18)                                            DMPIIT      61    A
DMPIIT     16       FORMAT(Z8*  PI WORD*Z3,3Z6/)                             DMPIIT      62    A
DMPIIT                                                                       DMPIIT      63    A
DMPIIT     20       ITW = IIT(I)                                             DMPIIT      64    A
DMPIIT              NW = NWD(ITW)                                            DMPIIT      65    A
DMPIIT              PRINT 21,NW,CA(ITW),UDI(ITW),NLD(IIT(I+1)),ILD(IIT(I+1))  DMPIIT     66    A
DMPIIT     21       FORMAT(10X,*NWD,CA,UDI =*2Z5,I5*  N.LD,ILD =*2Z5/)       DMPIIT      67    A
DMPIIT              J = I + NW                                               DMPIIT      68    A
DMPIIT              I = I + 2                                                DMPIIT      69    A
DMPIIT                                                                       DMPIIT      70    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DMPIIT

```
DMPIIT    30       PRINT 31                                                    DMPIIT        71     A
DMPIIT    31       FORMAT(10X)                                                 DMPIIT        72     A
DMPIIT             CALL PRNTRLI( IIT(I) )                                      DMPIIT        73     A
DMPIIT             I = I + 4                                                   DMPIIT        74     A
DMPIIT             IF( I .LT. J ) GO TO 30                                     DMPIIT        75     A
DMPIIT             IF( IPASS .NE. 0 ) GO TO 35                                 DMPIIT        76     A
DMPIIT             IPASS = 1                                                   DMPIIT        77     A
DMPIIT             GO TO 20                                                    DMPIIT        78     A
DMPIIT                                                                         DMPIIT        79     A
DMPIIT    35       IF( IIT(I) .NE. 0 ) GO TO 15                                DMPIIT        80     A
DMPIIT                                                                         DMPIIT        81     A
DMPIIT             LINECM = 0                                                  DMPIIT        82     A
DMPIIT             RETURN                                                      DMPIIT        83     A
DMPIIT                                                                         DMPIIT        84     A
DMPIIT                                                                         DMPIIT        85     A
DMPIIT             ENTRY  PRNTRXI                                              DMPIIT        86     A
DMPIIT                                                                         DMPIIT        87     A
DMPIIT   *         PRNT RXI - PRINT OUT REGION EXIT INFORMATION ( GPO/FXI )    DMPIIT        88     A
DMPIIT                                                                         DMPIIT        89     A
DMPIIT             I = 2                                                       DMPIIT        90     A
DMPIIT                                                                         DMPIIT        91     A
DMPIIT    100      X = IIT(I)                                                  DMPIIT        92     A
DMPIIT             PRINT 101,SCR(X),JP(X),ALLP(X),PRED(X),BI(X)                DMPIIT        93     A
DMPIIT    101      FORMAT(/*    SCR,JP,ALLP =*3Z2*   PRED,SUCC BN =*2Z5/)      DMPIIT        94     A
DMPIIT             I = I+1                                                     DMPIIT        95     A
DMPIIT             CALL PRNTABV( VECNAM , IIT(I) )                             DMPIIT        96     A
DMPIIT             I = I + VL                                                  DMPIIT        97     A
DMPIIT             IF( IIT(I) .NE. 0 ) GO TO 100                               DMPIIT        98     A
DMPIIT                                                                         DMPIIT        99     A
DMPIIT             RETURN                                                      DMPIIT       100     A
DMPIIT                                                                         DMPIIT       101     A
DMPIIT             END                                                         DMPIIT       102     A
```

SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - DMPIIT

IDENTIFIER	TOTAL	ACTIVE
DMPIIT	102	102

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DMPRLST

```
DMPRLST  *DECK     DMPRLST                                                     DMPRLST        1     A
DMPRLST            SUBROUTINE DMPRLST( CTR , TXT , LTXT )                      DMPRLST        2     A
DMPRLST                                                                        DMPRLST        3     A
DMPRLST  **        DMP RLST - PRINT *IL* INSTRUCTIONS IN *TXT* TABLE           DMPRLST        4     A
DMPRLST                                                                        DMPRLST        5     A
DMPRLST            IMPLICIT INTEGER(A-Z)                                       DMPRLST        6     A
DMPRLST            INTEGER TXT(LTXT) , CTR(5) , STPR(4) , ZPT(2)               DMPRLST        7     A
DMPRLST            COMMON/LINE/LINECM(2)                                       DMPRLST        8     A
DMPRLST            DATA ZPT / 2R    , 2R Z /                                   DMPRLST        9     A
DMPRLST            DATA STPR / 3R     , 3R RS   , 3R SP   , 3R **  /           DMPRLST       10     A
DMPRLST            FIELD(X,P,L) = SHIFT(X,60-P).A. .N.MASK(60-L)               DMPRLST       11     A
```

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DMPRLST

```
DMPRLST            ZP(X) = FIELD(X,32B,1)                                   DMPRLST      12     A
DMPRLST            PRS(X) = FIELD(X,38,2)                                   DMPRLST      13     A
DMPRLST            OC(X) = FIELD(X,48,10)                                   DMPRLST      14     A
DMPRLST                                                                     DMPRLST      15     A
DMPRLST            CTR(1) = CTR(1)+1                                        DMPRLST      16     A
DMPRLST            IF( CTR(1) .LT. CTR(3) ) RETURN                          DMPRLST      17     A
DMPRLST            IF( CTR(3) .GT. CTR(4) ) RETURN                          DMPRLST      18     A
DMPRLST            CTR(3) = CTR(3)+CTR(5)                                   DMPRLST      19     A
DMPRLST            CALL SETST( 1 , 8L            )                          DMPRLST      20     A
DMPRLST            PRINT 1,(CTR(I),I=1,2),LOCF(TXT)                         DMPRLST      21     A
DMPRLST   1        FORMAT(//*   CALL*I5* OF -IL- DUMP FROM -*A10*   FWA =*Z6  DMPRLST      22     A
DMPRLST       .    /13X,*LINK WORD   INDEX  DESCRIPTOR*)                    DMPRLST      23     A
DMPRLST                                                                     DMPRLST      24     A
DMPRLST            LINECM = LINECM(2)                                       DMPRLST      25     A
DMPRLST            N = LTXT                                                 DMPRLST      26     A
DMPRLST            I = 1                                                    DMPRLST      27     A
DMPRLST                                                                     DMPRLST      28     A
DMPRLST   10       DESCR = TXT(I+2)                                         DMPRLST      29     A
DMPRLST            PSB = STPR( PRS(DESCR)+1 )                               DMPRLST      30     A
DMPRLST            ZPB = ZPT( ZP(DESCR) + 1 )                               DMPRLST      31     A
DMPRLST                                                                     DMPRLST      32     A
DMPRLST            LW = TXT(I+3)                                            DMPRLST      33     A
DMPRLST            UB = FIELD(LW,36,24)                                     DMPRLST      34     A
DMPRLST            MB = FIELD(LW,18,18)                                     DMPRLST      35     A
DMPRLST            J = I - 1                                                DMPRLST      36     A
DMPRLST            UBD = FIELD(DESCR,18,42)                                 DMPRLST      37     A
DMPRLST            MBD = FIELD(DESCR,9,9)                                   DMPRLST      38     A
DMPRLST   C                                PRINT THE LINK WORD              DMPRLST      39     A
DMPRLST            PRINT  11,UB,MB,LW,J                                     DMPRLST      40     A
DMPRLST    11      FORMAT(2X,O8,2Z6,Z8)                                     DMPRLST      41     A
DMPRLST   C                                PRINT THE DESCRIPTOR WORD        DMPRLST      42     A
DMPRLST            IF( DESCR .NE. 0 ) GO TO 16                              DMPRLST      43     A
DMPRLST            PRINT 12,DEXCR                                           DMPRLST      44           I
          -CCG11
DMPRLST            PRINT 12,DESCR                                           CCG11       139     A
DMPRLST    12      FORMAT(I4,23X)                                           DMPRLST      45     A
DMPRLST            GO TO 18                                                 DMPRLST      46     A
DMPRLST                                                                     DMPRLST      47     A
DMPRLST    16      PRINT 17,UBD,MBD,DESCR,ZPB,PSB                           DMPRLST      48     A
DMPRLST    17      FORMAT(2X,O14,2Z3,R2,R3)                                 DMPRLST      49     A
DMPRLST   C                                PRINT THE INSTRUCTION - OPCD  RI,...  DMPRLST  50     A
DMPRLST    18      CALL PRNTRLI( TXT(I) )                                   DMPRLST      51     A
DMPRLST            I = I + 4                                                DMPRLST      52     A
DMPRLST            IF( I .GT. N ) GO TO 50                                  DMPRLST      53     A
DMPRLST            IF( OC( TXT(I-4) ) .NE. 0 ) GO TO 10                     DMPRLST      54     A
DMPRLST                                                                     DMPRLST      55     A
DMPRLST   *        PRINT TRAILING ENTRIES IN OCTAL                          DMPRLST      56     A
DMPRLST                                                                     DMPRLST      57     A
DMPRLST            PRINT 21                                                 DMPRLST      58     A
DMPRLST    21      FORMAT(/*  -- TRAILER WORDS --*//)                       DMPRLST      59     A
DMPRLST                                                                     DMPRLST      60     A
DMPRLST    20      PRINT 22,TXT(I)                                          DMPRLST      61     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DMPRLST

DMPRLST   22       FORMAT(10X,O20/)                                          DMPRLST       62     A
DMPRLST            I = I + 1                                                 DMPRLST       63     A
DMPRLST            IF( I .LE. N ) GO TO 20                                   DMPRLST       64     A
DMPRLST                                                                      DMPRLST       65     A
DMPRLST   50       LINECM = 0                                                DMPRLST       66     A
DMPRLST            RETURN                                                    DMPRLST       67     A
DMPRLST            END                                                       DMPRLST       68     A

          SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - DMPRLST

                IDENTIFIER                TOTAL    ACTIVE

                DMPRLST                      68        67
                CCG11                         1         1

                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DMPSIT

DMPSIT    *DECK    DMPSIT                                                    DMPSIT         1     A
DMPSIT             SUBROUTINE DMP SIT( CTR , TREE , LTREE , TXT )           DMPSIT         2     A
DMPSIT                                                                       DMPSIT         3     A
DMPSIT    **       DMP SIT - PRINT SUCCESSOR INDEX TABLE ( BDT EXIT )       DMPSIT         4     A
DMPSIT                                                                       DMPSIT         5     A
DMPSIT             IMPLICIT INTEGER(A-Z)                                    DMPSIT         6     A
DMPSIT             INTEGER TREE(LTREE),TXT(1),CTR(5),SUBTTL(6)              DMPSIT         7     A
DMPSIT             COMMON/LINE/LINECM(2)                                    DMPSIT         8     A
DMPSIT             DATA SUBTTL(1) /                                         DMPSIT         9     A
DMPSIT        .  48L     LST    PRED  SUCC  T(S)  USES  N.PR  INST        /  DMPSIT        10     A
DMPSIT                                                                       DMPSIT        11     A
DMPSIT             FIELD(X,P,L) = SHIFT(X,60-P).A. .N.MASK(60-L)            DMPSIT        12     A
DMPSIT             PRED(X) = FIELD(X,36,18)                                 DMPSIT        13     A
DMPSIT             ULST(X) = FIELD(X,18,18)                                 DMPSIT        14     A
DMPSIT             SUCC(X) = FIELD(X,0,18)                                  DMPSIT        15     A
DMPSIT                                                                       DMPSIT        16     A
DMPSIT             USES(X) = FIELD(X,48,9)                                  DMPSIT        17     A
DMPSIT             LST(X) = FIELD(X,18,15)                                  DMPSIT        18     A
DMPSIT             NPRED(X) = FIELD(X,0,18)                                 DMPSIT        19     A
DMPSIT                                                                       DMPSIT        20     A
DMPSIT             CTR = CTR + 1                                            DMPSIT        21     A
DMPSIT             LN = FIELD(TXT,30,18)                                    DMPSIT        22     A
DMPSIT             PRINT 5,CTR(1),LN                                        DMPSIT        23     A
DMPSIT    5        FORMAT(//* CALL*I5* TO TREE DMP, LINE NUMBER =*I5)       DMPSIT        24     A
DMPSIT             CALL SETST( 5 , SUBTTL )                                 DMPSIT        25     A
DMPSIT                                                                       DMPSIT        26     A
DMPSIT             LINECM = LINECM(2)                                       DMPSIT        27     A
DMPSIT             PLAST = -1                                               DMPSIT        28     A
DMPSIT                                                                       DMPSIT        29     A
DMPSIT             DO 30 J = 1,LTREE                                        DMPSIT        30     A
DMPSIT             S = SUCC( TREE(J) )                                      DMPSIT        31     A
DMPSIT             TS = ULST( TREE(J) )                                     DMPSIT        32     A
DMPSIT             P = PRED( TREE(J) )                                      DMPSIT        33     A
DMPSIT             IF( P .EQ. PLAST ) GO TO 20                              DMPSIT        34     A
DMPSIT                                                                       DMPSIT        35     A

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DMPSIT

      DMPSIT              PLAST = P                                                 DMPSIT        36     A
      DMPSIT              LW = TXT(P+4)                                             DMPSIT        37     A
      DMPSIT              RK = LST(LW)                                              DMPSIT        38     A
      DMPSIT              U = USES(LW)                                              DMPSIT        39     A
      DMPSIT              NP = NPRED(LW)                                            DMPSIT        40     A
      DMPSIT              PRINT 15,RK,P,S,TS,U,NP                                   DMPSIT        41     A
      DMPSIT    15        FORMAT(I8,Z8,Z6,3I6,2X)                                   DMPSIT        42     A
      DMPSIT              CALL PRNT RLI( TXT(P+1) )                                 DMPSIT        43     A
      DMPSIT              GO TO 30                                                  DMPSIT        44     A
      DMPSIT                                                                        DMPSIT        45     A
      DMPSIT    20        PRINT 25,S,TS                                             DMPSIT        46     A
      DMPSIT    25        FORMAT(16X,Z6,I6/)                                        DMPSIT        47     A
      DMPSIT                                                                        DMPSIT        48     A
      DMPSIT    30        CONTINUE                                                  DMPSIT        49     A
      DMPSIT                                                                        DMPSIT        50     A
      DMPSIT              LINECM = 0                                                DMPSIT        51     A
      DMPSIT              RETURN                                                    DMPSIT        52     A
      DMPSIT              END                                                       DMPSIT        53     A

           SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - DMPSIT

                IDENTIFIER                TOTAL    ACTIVE

                DMPSIT                       53        53

                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DMPTREE

      DMPTREE  *DECK      DMPTREE                                                   DMPTREE        1     A
      DMPTREE             SUBROUTINE DMPTREE( CTR, TREE , LTREE , TXT )             DMPTREE        2     A
      DMPTREE                                                                       DMPTREE        3     A
      DMPTREE  **         DMP TREE - PRINT DEPENDENCY TREE ( BDT/CIP )              DMPTREE        4     A
      DMPTREE                                                                       DMPTREE        5     A
      DMPTREE             IMPLICIT INTEGER(A-Z)                                     DMPTREE        6     A
      DMPTREE             INTEGER TREE(LTREE),TXT(1),CTR(5),SUBTTL(6)               DMPTREE        7     A
      DMPTREE             COMMON/LINE/LINECM(2)                                     DMPTREE        8     A
      DMPTREE             COMMON/MCG/XX(2),SEQTIME                                  DMPTREE        9     A
      DMPTREE             DATA SUBTTL(1) /                                          DMPTREE       10     A
      DMPTREE        .  58L     LST    TIME    SUCC  PRED  USES  N.PR  INST     /   DMPTREE       11     A
      DMPTREE                                                                       DMPTREE       12     A
      DMPTREE             FIELD(X,P,L) = SHIFT(X,60-P).A. .N.MASK(60-L)             DMPTREE       13     A
      DMPTREE             SUCCO(X) = FIELD(X,24,18)                                 DMPTREE       14     A
      DMPTREE             SUCC(X) = FIELD(X,26,16)+1                                DMPTREE       15     A
      DMPTREE             PRED(X) = FIELD(X,2,16)+1                                 DMPTREE       16     A
      DMPTREE                                                                       DMPTREE       17     A
      DMPTREE             USES(X) = FIELD(X,48,9)                                   DMPTREE       18     A
      DMPTREE             LST(X) = FIELD(X,18,15)                                   DMPTREE       19     A
      DMPTREE             NPRED(X) = FIELD(X,0,18)                                  DMPTREE       20     A
      DMPTREE                                                                       DMPTREE       21     A
      DMPTREE             CTR(1) = CTR(1) + 1                                       DMPTREE       22     A
      DMPTREE             LN = FIELD(TXT,30,18)                                     DMPTREE       23     A
      DMPTREE             PRINT 5,CTR(1),LN                                         DMPTREE       24     A
      DMPTREE    5        FORMAT(//*  CALL*I5* TO TREE DMP,LINE NUMBER = *I5)       DMPTREE       25     A

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DMPTREE

```
DMPTREE            CALL SETST( 6 , SUBTTL )                                  DMPTREE       26     A
DMPTREE            LINECM = LINECM(2)                                        DMPTREE       27     A
DMPTREE                                                                      DMPTREE       28     A
DMPTREE            SLAST = 0                                                 DMPTREE       29     A
DMPTREE            DO 30 J = 1,LTREE                                         DMPTREE       30     A
DMPTREE            S = SUCC( TREE(J) )                                       DMPTREE       31     A
DMPTREE            P = PRED( TREE(J) )                                       DMPTREE       32     A
DMPTREE            IF( S .EQ. SLAST ) GO TO 20                               DMPTREE       33     A
DMPTREE                                                                      DMPTREE       34     A
DMPTREE            SLAST = S                                                 DMPTREE       35     A
DMPTREE            SO = SUCCO( TREE(J) )                                     DMPTREE       36     A
DMPTREE            LW = TXT( SO + 4 )                                        DMPTREE       37     A
DMPTREE            RK = LST(LW)                                              DMPTREE       38     A
DMPTREE            PR = SEQTIME - RK                                         DMPTREE       39     A
DMPTREE            U = USES(LW)                                              DMPTREE       40     A
DMPTREE            NP = NPRED(LW)                                            DMPTREE       41     A
DMPTREE            PRINT 15,RK,PR,S,P,U,NP                                   DMPTREE       42     A
DMPTREE    15      FORMAT(3I8,3I6,2X)                                        DMPTREE       43     A
DMPTREE            CALL PRNT RLI( TXT( SO + 1 ) )                            DMPTREE       44     A
DMPTREE            GO TO 30                                                  DMPTREE       45     A
DMPTREE                                                                      DMPTREE       46     A
DMPTREE    20      PRINT 25,P                                                DMPTREE       47     A
DMPTREE    25      FORMAT(24X,I6/)                                           DMPTREE       48     A
DMPTREE                                                                      DMPTREE       49     A
DMPTREE    30      CONTINUE                                                  DMPTREE       50     A
DMPTREE                                                                      DMPTREE       51     A
DMPTREE            LINECM = 0                                                DMPTREE       52     A
DMPTREE            RETURN                                                    DMPTREE       53     A
DMPTREE            END                                                       DMPTREE       54     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - DMPTREE

         IDENTIFIER                TOTAL    ACTIVE

         DMPTREE                      54        54

         LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DMPUDI

DMPUDI     *DECK    DMPUDI                                                    DMPUDI         1     A
DMPUDI              SUBROUTINE DMPUDI( UDI , N )                              DMPUDI         2     A
DMPUDI                                                                        DMPUDI         3     A
DMPUDI     **       DMPUDI - PRINT ENTRIES IN *UDI* BEFORE THEY ARE REFORMATTED DMPUDI       4     A
DMPUDI                                                                        DMPUDI         5     A
DMPUDI              IMPLICIT INTEGER(A-Z)                                     DMPUDI         6     A
DMPUDI              INTEGER UDI(N)                                            DMPUDI         7     A
DMPUDI              COMMON/LINE/LINECM(2)                                     DMPUDI         8     A
DMPUDI                                                                        DMPUDI         9     A
DMPUDI              FIELD(X,P,L) = SHIFT(X,60-P) .A. .N.MASK(60-L)            DMPUDI        10     A
DMPUDI              IH(X) = FIELD(X,0,18)                                     DMPUDI        11     A
DMPUDI              CA(X) = FIELD(X,18,18)                                    DMPUDI        12     A
DMPUDI              CR(X) = FIELD(X,54,1)                                     DMPUDI        13     A
DMPUDI                                                                        DMPUDI        14     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DMPUDI

```
      DMPUDI              DATA NN / 0 /                                                  DMPUDI        15     A
      DMPUDI                                                                             DMPUDI        16     A
      DMPUDI              PRINT 1                                                        DMPUDI        17     A
      DMPUDI     1        FORMAT(///*  -- UDI SYMBOL DICTIONARY --*//)                   DMPUDI        18     A
      DMPUDI     3        FORMAT(I6,4X)                                                  DMPUDI        19     A
      DMPUDI     2        FORMAT(/)                                                      DMPUDI        20     A
      DMPUDI     4        FORMAT(4X,Z6,2X,A7)                                            DMPUDI        21     A
      DMPUDI     5        FORMAT(12H   -CLASS-   ,A7)                                    DMPUDI        22     A
      DMPUDI                                                                             DMPUDI        23     A
      DMPUDI              LINECM = LINECM(2)                                             DMPUDI        24     A
      DMPUDI              K = (N-2)/4                                                    DMPUDI        25     A
      DMPUDI              I = 3                                                          DMPUDI        26     A
      DMPUDI                                                                             DMPUDI        27     A
      DMPUDI     10       J = I - 1                                                      DMPUDI        28     A
      DMPUDI              PRINT 3, J                                                     DMPUDI        29     A
      DMPUDI              J = MIN0( N , I+2 )                                            DMPUDI        30     A
      DMPUDI                                                                             DMPUDI        31     A
      DMPUDI              DO 20 K = I,J                                                  DMPUDI        32     A
      DMPUDI              IHV = SYM( IH( UDI(K) ) )                                      DMPUDI        33     A
      DMPUDI              CAV = CA( UDI(K) )                                             DMPUDI        34     A
      DMPUDI              IF( CR( UDI(K) ) .NE. 0 ) GO TO 15                             DMPUDI        35     A
      DMPUDI              IF( SHIFT(CAV,42) .LT. 0 ) CAV = CAV .O. MASK(42)              DMPUDI        36     A
      DMPUDI              PRINT 4,CAV,IHV                                                DMPUDI        37     A
      DMPUDI              GO TO 20                                                       DMPUDI        38     A
      DMPUDI     15       PRINT 5,IHV                                                    DMPUDI        39     A
      DMPUDI     20       CONTINUE                                                       DMPUDI        40     A
      DMPUDI              PRINT 2                                                        DMPUDI        41     A
      DMPUDI                                                                             DMPUDI        42     A
      DMPUDI              I = I + 3                                                      DMPUDI        43     A
      DMPUDI              IF( I .LE. N ) GO TO 10                                        DMPUDI        44     A
      DMPUDI                                                                             DMPUDI        45     A
      DMPUDI              IF( NN .NE. 0 ) RETURN                                         DMPUDI        46     A
      DMPUDI                                                                             DMPUDI        47     A
      DMPUDI    *         PRINT THE NAMES IN THE SYMBOL TABLE, 8 NAMES PER LINE          DMPUDI        48     A
      DMPUDI                                                                             DMPUDI        49     A
      DMPUDI              PRINT 25                                                       DMPUDI        50     A
      DMPUDI     25       FORMAT(///*  -- SYMBOL TABLE NAMES --*//)                      DMPUDI        51     A
      DMPUDI              CALL GETNSYM( NN )                                             DMPUDI        52     A
      DMPUDI              J = 1                                                          DMPUDI        53     A
      DMPUDI                                                                             DMPUDI        54     A
      DMPUDI     30       PRINT 31,J                                                     DMPUDI        55     A
      DMPUDI     31       FORMAT(Z6)                                                     DMPUDI        56     A
      DMPUDI              K = MIN0( I+8 , NN )                                           DMPUDI        57           I
               -CCG27
      DMPUDI              K = MIN0( J+8 , NN )                                           CCG27         69     A
      DMPUDI     40       IHV = SYM(J)                                                   DMPUDI        58     A
      DMPUDI              PRINT 41, IHV                                                  DMPUDI        59     A
      DMPUDI     41       FORMAT(2X,A8)                                                  DMPUDI        60     A
      DMPUDI              J = J + 1                                                      DMPUDI        61     A
      DMPUDI              IF( J .LE. K ) GO TO 40                                        DMPUDI        62     A
      DMPUDI                                                                             DMPUDI        63     A
      DMPUDI              PRINT 2                                                        DMPUDI        64     A

               0         1         2         3         4         5         6         7         8
               12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN DMPUDI

      DMPUDI            IF( J .LT. NN ) GO TO 30                                DMPUDI        65     A
      DMPUDI                                                                    DMPUDI        66     A
      DMPUDI            LINECM = 0                                              DMPUDI        67     A
      DMPUDI            RETURN                                                  DMPUDI        68     A
      DMPUDI                                                                    DMPUDI        69     A
      DMPUDI            END                                                     DMPUDI        70     A

           SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - DMPUDI

                IDENTIFIER                TOTAL    ACTIVE

                DMPUDI                       70        69
                CCG27                         1         1

                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PRNTGRA

      PRNTGRA  *DECK    PRNTGRA                                                 PRNTGRA        1     A
      PRNTGRA           SUBROUTINE PRNT GRA( NAME , RCT , LEN , INDX )          PRNTGRA        2     A
      PRNTGRA           IMPLICIT INTEGER(A-Z)                                   PRNTGRA        3     A
      PRNTGRA                                                                   PRNTGRA        4     A
      PRNTGRA  *        PRNT GRA - PRINT OUT TABLES USED BY *GRA*               PRNTGRA        5     A
      PRNTGRA                                                                   PRNTGRA        6     A
      PRNTGRA           INTEGER RCT(LEN) , BVAL(17) , RCTTTL(6)                 PRNTGRA        7     A
      PRNTGRA           COMMON/RCTFMT/BNAM(16) /LINE/LINECM(2)                  PRNTGRA        8     A
      PRNTGRA           COMMON/TABLES/ DUM(100B) , RAT(24),RVT(24) , IAI,TRA    PRNTGRA        9           I
               -CCG18
      PRNTGRA           COMMON/TABLES/ ERC(3) , NVC,TUD , ABR,NAB,NTA,TRD ,     PRNTGRA       10           I
               -CCG18
      PRNTGRA           COMMON/CCGSCR/ DUM(64) , RAT(24),RVT(24) , IAI,TRA      CCG18        176     A
      PRNTGRA           COMMON/CCGSCR/ ERC(3),NVC,TUD,ABR,NAB,NTA,TRD,          CCG18        177     A
      PRNTGRA       1          NXC,XCT(5),MLW,NLD(2) , MAA,NAA,ACT(8) , MFA(10) , SUPS  PRNTGRA       11     A
      PRNTGRA       2  , NXS,MRU,NPS,PST(13)                                    PRNTGRA       12     A
      PRNTGRA                                                                   PRNTGRA       13     A
      PRNTGRA           COMMON/GPOGRA/OMIT(10) , MXPITL,EMF,ITL,MRA,LGL,NHB,XRF ,  PRNTGRA      14     A
      PRNTGRA       1   LCV(2),LUL(2),LIH,LR2W,NLJ,LTT,OENL,NENL,EPSI,MAXW,NEBB,  PRNTGRA       15     A
      PRNTGRA       2          PITL,STP                                         PRNTGRA       16     A
      PRNTGRA                                                                   PRNTGRA       17     A
      PRNTGRA           DATA RCTTTL / 58L RCT  RF     CA  IH     H2    REG1,2,IT PFT A  PRNTGRA  18     A
      PRNTGRA       .PF NOC USE            /                                    PRNTGRA       19     A
      PRNTGRA                                                                   PRNTGRA       20     A
      PRNTGRA           FIELD(X,P,L) = .N.MASK(60-L).A.SHIFT(X,60-P)            PRNTGRA       21     A
      PRNTGRA  *                          CAW                                   PRNTGRA       22     A
      PRNTGRA           PRFT(X) = FIELD(X,44,3)                                 PRNTGRA       23     A
      PRNTGRA           APLN(X) = FIELD(X,42,2)                                 PRNTGRA       24     A
      PRNTGRA           NOCC(X) = FIELD(X,33,9)                                 PRNTGRA       25     A
      PRNTGRA           SUSE(X) = FIELD(X,24,9)                                 PRNTGRA       26     A
      PRNTGRA           UDI(X) = FIELD(X,0,18)                                  PRNTGRA       27     A
      PRNTGRA  *                          C2W                                   PRNTGRA       28     A
      PRNTGRA           MSK(X) = FIELD(X,59,1)                                  PRNTGRA       29     A
      PRNTGRA           RF(X)  = FIELD(X,36,18)                                 PRNTGRA       30     A
      PRNTGRA           CA(X)  = FIELD(X,18,18)                                 PRNTGRA       31     A
      PRNTGRA           IH(X)  = FIELD(X,0,12)                                  PRNTGRA       32     A

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PRNTGRA

```
PRNTGRA  *                                CCW                                    PRNTGRA    33    A
PRNTGRA            INCB(X) = FIELD(X,57,3)                                       PRNTGRA    34    A
PRNTGRA            INCP(X) = FIELD(X,36,18)                                      PRNTGRA    35    A
PRNTGRA            H2(X) = FIELD(X,18,18)                                        PRNTGRA    36    A
PRNTGRA            REG2(X) = FIELD(X,12,6)                                       PRNTGRA    37    A
PRNTGRA            MIT(X) = FIELD(X,6,6)                                         PRNTGRA    38    A
PRNTGRA            REG1(X) = FIELD(X,0,6)                                        PRNTGRA    39    A
PRNTGRA  *                                RAT                                    PRNTGRA    40    A
PRNTGRA            INV(X) = FIELD(X,59,1)                                        PRNTGRA    41    A
PRNTGRA            R3(X) = FIELD(X,36,18)                                        PRNTGRA    42    A
PRNTGRA            R2(X) = FIELD(X,18,18)                                        PRNTGRA    43    A
PRNTGRA            R1(X) = FIELD(X,0,18)                                         PRNTGRA    44    A
PRNTGRA                                                                          PRNTGRA    45    A
PRNTGRA            INDXL = INDX                                                  PRNTGRA    46    A
PRNTGRA            PRINT 1,NAME                                                  PRNTGRA    47    A
PRNTGRA    1       FORMAT(/*   PRNTGRA CALL FROM -*A10)                          PRNTGRA    48    A
PRNTGRA    2       FORMAT(/)                                                     PRNTGRA    49    A
PRNTGRA    3       FORMAT(6X)                                                    PRNTGRA    50    A
PRNTGRA    4       FORMAT(I4*U *)                                                PRNTGRA    51    A
PRNTGRA            IF( RCT(1) .EQ. 0 ) GO TO 10                                  PRNTGRA    52    A
PRNTGRA                                                                          PRNTGRA    53    A
PRNTGRA            CALL REMARK( 18L -- RCT ERROR -                )              PRNTGRA    54    A
PRNTGRA            PRINT 6,RCT(1)                                                PRNTGRA    55    A
PRNTGRA    6       FORMAT(//* ---- ERROR, RCT(0) = *O20* ----*//)                PRNTGRA    56    A
PRNTGRA                                                                          PRNTGRA    57    A
PRNTGRA    10      INDXL = SHIFT(INDXL,6)                                        PRNTGRA    58    A
PRNTGRA            I = INDXL .A. 77B                                             PRNTGRA    59    A
PRNTGRA            LINECM = 0                                                    PRNTGRA    60    A
PRNTGRA            IF( I .EQ. 0 ) RETURN                                         PRNTGRA    61    A
PRNTGRA                                                                          PRNTGRA    62    A
PRNTGRA  *                 RCT   RAT    RVT   SEE   MTA   DXA   DAA   MFA   SUP  PRNTGRA    63    A
PRNTGRA            GO TO ( 20 , 100 , 140 , 160 , 180 , 200 , 220 , 240 , 260 ,  PRNTGRA    64    A
PRNTGRA       1                        280 ) , I                                 PRNTGRA    65    A
PRNTGRA                                                                          PRNTGRA    66    A
PRNTGRA  *         PRINT CONTENTS OF *RCT*                                       PRNTGRA    67    A
PRNTGRA                                                                          PRNTGRA    68    A
PRNTGRA    20      PRINT 2                                                       PRNTGRA    69    A
PRNTGRA            CALL SETST( 6 , RCTTTL )                                      PRNTGRA    70    A
PRNTGRA            LINECM = LINECM(2)                                            PRNTGRA    71    A
PRNTGRA                                                                          PRNTGRA    72    A
PRNTGRA            RCT(1) = MASK(42)                                             PRNTGRA    73    A
PRNTGRA            DO 50 J = 1,LEN,3                                             PRNTGRA    74    A
PRNTGRA            CAW = RCT(J)                                                  PRNTGRA    75    A
PRNTGRA            C2W = RCT(J+1)                                                PRNTGRA    76    A
PRNTGRA            CCW = RCT(J+2)                                                PRNTGRA    77    A
PRNTGRA            IHV = IH(C2W)                                                 PRNTGRA    78    A
PRNTGRA            IF( IHV .EQ. 0 ) GO TO 23                                     PRNTGRA    79    A
PRNTGRA            SYV = SYM(IHV)                                                PRNTGRA    80    A
PRNTGRA            GO TO 24                                                      PRNTGRA    81    A
PRNTGRA    23      SYV = 7L                                                      PRNTGRA    82    A
PRNTGRA            IF( MSK(C2W) .EQ. 0 ) GO TO 24                                PRNTGRA    83    A
PRNTGRA            SYV = 7L -MSK-                                                PRNTGRA    84    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PRNTGRA

```
PRNTGRA                                                                  PRNTGRA     85    A
PRNTGRA   24       PRINT 21, J-1 , RF(C2W),CA(C2W),SYV                   PRNTGRA     86    A
PRNTGRA   21       FORMAT(1X,I3,I4,Z7,1X,A7)                             PRNTGRA     87    A
PRNTGRA                                                                  PRNTGRA     88    A
PRNTGRA            SYV = SYM( H2(CCW) )                                  PRNTGRA     89    A
PRNTGRA                                                                  PRNTGRA     90    A
PRNTGRA            PRINT 25,SYV,REG1(CCW),REG2(CCW),MIT(CCW)             PRNTGRA     91    A
PRNTGRA   25       FORMAT(1X,A7,Z3,Z2,Z3)                                PRNTGRA     92    A
PRNTGRA            PRINT 26,PRFT(CAW),APLN(CAW),NOCC(CAW),SUSE(CAW)      PRNTGRA     93    A
PRNTGRA   26       FORMAT(4I4)                                           PRNTGRA     94    A
PRNTGRA                                                                  PRNTGRA     95    A
PRNTGRA            IF( INCP(CCW) .EQ. 0 ) GO TO 40                       PRNTGRA     96    A
PRNTGRA            PRINT 27,INCB(CCW),INCP(CCW)                          PRNTGRA     97    A
PRNTGRA   27       FORMAT(Z2,I4)                                         PRNTGRA     98    A
PRNTGRA            GO TO 41                                              PRNTGRA     99    A
PRNTGRA                                                                  PRNTGRA    100    A
PRNTGRA   40       PRINT 3                                               PRNTGRA    101    A
PRNTGRA                                                                  PRNTGRA    102    A
PRNTGRA   41       U = UDI(CAW)/2                                        PRNTGRA    103    A
PRNTGRA            IF( U.NE. 0 ) GO TO 28                                PRNTGRA    104    A
PRNTGRA            PRINT 3                                               PRNTGRA    105    A
PRNTGRA            GO TO 29                                              PRNTGRA    106    A
PRNTGRA   28       PRINT 4,U                                             PRNTGRA    107    A
PRNTGRA                                                                  PRNTGRA    108    A
PRNTGRA  *         COLLECT BIT VALUES AND PRINT THEM                     PRNTGRA    109    A
PRNTGRA                                                                  PRNTGRA    110    A
PRNTGRA   29       DO 30 I = 1,17                                        PRNTGRA    111    A
PRNTGRA            BVAL(I) = 5R                                          PRNTGRA    112    A
PRNTGRA            IF( (SHIFT(CAW,BNAM(I)).A.1) .EQ. 0 ) GO TO 30        PRNTGRA    113    A
PRNTGRA            BVAL(I) = SHIFT( MASK(42).A.BNAM(I) , -18 )           PRNTGRA    114    A
PRNTGRA   30       CONTINUE                                              PRNTGRA    115    A
PRNTGRA                                                                  PRNTGRA    116    A
PRNTGRA            PRINT 31,BVAL                                         PRNTGRA    117    A
PRNTGRA   31       FORMAT(3R3,2(R4,3R3),3R4,2R3,R4/)                     PRNTGRA    118    A
PRNTGRA                                                                  PRNTGRA    119    A
PRNTGRA   50       CONTINUE                                              PRNTGRA    120    A
PRNTGRA                                                                  PRNTGRA    121    A
PRNTGRA            RCT(1) = 0                                            PRNTGRA    122    A
PRNTGRA            CALL SETST( 0 )                                       PRNTGRA    123    A
PRNTGRA            GO TO 10                                              PRNTGRA    124    A
PRNTGRA                                                                  PRNTGRA    125    A
PRNTGRA  *         PRINT NON-ZERO LOCATIONS IN *RAT*                     PRNTGRA    126    A
PRNTGRA                                                                  PRNTGRA    127    A
PRNTGRA   100      PRINT 101                                             PRNTGRA    128    A
PRNTGRA   101      FORMAT(/* REGNO INV  R1  R2  R3*)                     PRNTGRA    129    A
PRNTGRA                                                                  PRNTGRA    130    A
PRNTGRA            DO 110 I = 1,24                                       PRNTGRA    131    A
PRNTGRA            IF( RAT(I) .EQ. 0 ) GO TO 110                         PRNTGRA    132    A
PRNTGRA            R = RAT(I)                                            PRNTGRA    133    A
PRNTGRA            PRINT 105,I-1,INV(R),R1(R),R2(R),R3(R)                PRNTGRA    134    A
PRNTGRA   105      FORMAT(Z6,Z4,3I4)                                     PRNTGRA    135    A
PRNTGRA                                                                  PRNTGRA    136    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PRNTGRA

```
PRNTGRA   110      CONTINUE                                                  PRNTGRA      137     A
PRNTGRA            GO TO 10                                                  PRNTGRA      138     A
PRNTGRA                                                                      PRNTGRA      139     A
PRNTGRA   *        PRINT CONTENTS OF *RVT*                                   PRNTGRA      140     A
PRNTGRA                                                                      PRNTGRA      141     A
PRNTGRA   140      PRINT 141,(RVT(I),I = 2,24)                               PRNTGRA      142     A
PRNTGRA   141      FORMAT(* RVT -*23Z5)                                      PRNTGRA      143     A
PRNTGRA            GO TO 10                                                  PRNTGRA      144     A
PRNTGRA                                                                      PRNTGRA      145     A
PRNTGRA                                                                      PRNTGRA      146     A
PRNTGRA   *                                     SEE                          PRNTGRA      147     A
PRNTGRA   160      XRFLG = XRF                                               PRNTGRA      148     A
PRNTGRA            IF( XRF .LT. 0 ) XRFLG = -1                               PRNTGRA      149     A
PRNTGRA            IF( STP .NE. 0 ) STP = 1                                  PRNTGRA      150     A
PRNTGRA            NLJJ = NLJ .A. 77777B                                     PRNTGRA      151     A
PRNTGRA            PRINT 161,MRA,NHB,XRFLG,NENL,EPSI,MAXW,NEBB,NVC,STP,NLJJ,LTT,  PRNTGRA  152     A
PRNTGRA       1          TUD,LCV(1),LUL(1)                                   PRNTGRA      153     A
PRNTGRA   161      FORMAT(/* SEE - MRA N.HB  XRF N.XN  EPSI MAXW N.EB N.VC  STP N  PRNTGRA  154     A
PRNTGRA       1.LJ  LTT  TUD*/Z10,3I5,Z6,7I5/*  LCV,LUL =*2O22)               PRNTGRA      155     A
PRNTGRA            GO TO 10                                                  PRNTGRA      156     A
PRNTGRA                                                                      PRNTGRA      157     A
PRNTGRA   *                                     MTA                          PRNTGRA      158     A
PRNTGRA   180      PRINT 181,TRD,ABR                                         PRNTGRA      159     A
PRNTGRA   181      FORMAT(/* MTA - TRD,ABR =*I3,2X,Z3)                       PRNTGRA      160     A
PRNTGRA            GO TO 10                                                  PRNTGRA      161     A
PRNTGRA                                                                      PRNTGRA      162     A
PRNTGRA   *                                     DXA                          PRNTGRA      163     A
PRNTGRA   200      PRINT 201,MLW,NLD,NXC,XCT                                 PRNTGRA      164     A
PRNTGRA   201      FORMAT(/* DXA - MLW N.LD N.ST  NXC  -XCT-*/I10,9I5)       PRNTGRA      165     A
PRNTGRA            GO TO 10                                                  PRNTGRA      166     A
PRNTGRA                                                                      PRNTGRA      167     A
PRNTGRA   *                                     DAA                          PRNTGRA      168     A
PRNTGRA   220      IF( NAA .EQ. 0 ) GO TO 10                                 PRNTGRA      169     A
PRNTGRA            LINECM = LINECM(2)                                        PRNTGRA      170     A
PRNTGRA            PRINT 221,MAA,NAA                                         PRNTGRA      171     A
PRNTGRA   221      FORMAT(/* DAA - MAX,N.AA = *2I3/*  A-CANIDATES, INC,ORD -*)  PRNTGRA  172     A
PRNTGRA                                                                      PRNTGRA      173     A
PRNTGRA            DO 230 I = 1,NAA                                          PRNTGRA      174     A
PRNTGRA            X = ACT(I)                                                PRNTGRA      175     A
PRNTGRA            PRINT 223,FIELD(X,57,3),FIELD(X,36,18),FIELD(X,0,18)      PRNTGRA      176     A
PRNTGRA   223      FORMAT(Z6,I3* X*Z1)                                       PRNTGRA      177     A
PRNTGRA   230      CONTINUE                                                  PRNTGRA      178     A
PRNTGRA            PRINT 2                                                   PRNTGRA      179     A
PRNTGRA            GO TO 10                                                  PRNTGRA      180     A
PRNTGRA                                                                      PRNTGRA      181     A
PRNTGRA   *                                     MFA                          PRNTGRA      182     A
PRNTGRA   240      GO TO 10                                                  PRNTGRA      183     A
PRNTGRA                                                                      PRNTGRA      184     A
PRNTGRA   *                                     SUP                          PRNTGRA      185     A
PRNTGRA   260      IF( NPS .EQ. 0 ) GO TO 10                                 PRNTGRA      186     A
PRNTGRA            PRINT 261                                                 PRNTGRA      187     A
PRNTGRA   261      FORMAT(* POST STORES, REG RCT ORD*/)                      PRNTGRA      188     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PRNTGRA

PRNTGRA                                                                  PRNTGRA      189     A
PRNTGRA            LINECM = LINECM(2)                                    PRNTGRA      190     A
PRNTGRA            DO 270 I = 1,NPS                                      PRNTGRA      191     A
PRNTGRA            X = PST(I)                                            PRNTGRA      192     A
PRNTGRA            PRINT 262,FIELD(X,18,18),FIELD(X,0,18)                PRNTGRA      193     A
PRNTGRA   262      FORMAT(Z5,I4)                                         PRNTGRA      194     A
PRNTGRA   270      CONTINUE                                              PRNTGRA      195     A
PRNTGRA            PRINT 2                                               PRNTGRA      196     A
PRNTGRA            GO TO 10                                              PRNTGRA      197     A
PRNTGRA                                                                  PRNTGRA      198     A
PRNTGRA  *                                       SXC                     PRNTGRA      199     A
PRNTGRA   280      GO TO 10                                              PRNTGRA      200     A
PRNTGRA                                                                  PRNTGRA      201     A
PRNTGRA            END                                                   PRNTGRA      202     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - PRNTGRA

          IDENTIFIER                TOTAL    ACTIVE

          PRNTGRA                     202       200
          CCG18                         2         2

          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PRNTMIO

PRNTMIO  *DECK     PRNTMIO                                               PRNTMIO        1     A
PRNTMIO        SUBROUTINE PRNTMIO(LAB,INDX,OBIT,LBIT,OBST,LBST,OBLK,LBLK,RL,MPW)  PRNTMIO        2     A
PRNTMIO        IMPLICIT INTEGER (A-Z)                                    PRNTMIO        3     A
PRNTMIO        DIMENSION OBLK(LBLK),OBIT(LBIT),OBST(LBST),CTR(5)         PRNTMIO        4     A
PRNTMIO        COMMON/LINE/LINECM(2)                                     PRNTMIO        5     A
PRNTMIO        DATA BLANK/10H          /,TEST/0/                         PRNTMIO        6     A
PRNTMIO        DATA NRL/1/                                               PRNTMIO        7     A
PRNTMIO                                                                  PRNTMIO        8     A
PRNTMIO  *         FIELD DEFINITIONS                                     PRNTMIO        9     A
PRNTMIO                                                                  PRNTMIO       10     A
PRNTMIO        FIELD(X,P,L) = SHIFT(X,60-P).A. .N. MASK(60-L)            PRNTMIO       11     A
PRNTMIO                                                                  PRNTMIO       12     A
PRNTMIO  *         RANDOM INDEX WORD                                     PRNTMIO       13     A
PRNTMIO                                                                  PRNTMIO       14     A
PRNTMIO        RIC(X) = FIELD(X,59,1)                                    PRNTMIO       15     A
PRNTMIO        RAT(X) = FIELD(X,58,1)                                    PRNTMIO       16     A
PRNTMIO        ROFS(X) = FIELD(X,30,9)                                   PRNTMIO       17     A
PRNTMIO        RLEN(X) = FIELD(X,39,18)                                  PRNTMIO       18     A
PRNTMIO        RRA(X) = FIELD(X,0,30)                                    PRNTMIO       19     A
PRNTMIO        RFWA(X) = FIELD(X,18,18)                                  PRNTMIO       20     A
PRNTMIO        RLCM(X) = FIELD(X,0,18)                                   PRNTMIO       21     A
PRNTMIO                                                                  PRNTMIO       22     A
PRNTMIO  *         BLOCK ADDRESS WORD                                    PRNTMIO       23     A
PRNTMIO                                                                  PRNTMIO       24     A
PRNTMIO        ABLK(X) = FIELD(X,59,1)                                   PRNTMIO       25     A
PRNTMIO        APRI(X) = FIELD(X,58,1)                                   PRNTMIO       26     A
PRNTMIO        AHB(X) = FIELD(X,57,1)                                    PRNTMIO       27     A
PRNTMIO        AFW(X) = FIELD(X,56,1)                                    PRNTMIO       28     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PRNTMIO

```
PRNTMIO        ABI(X) = FIELD(X,36,18)                                  PRNTMIO      29    A
PRNTMIO        ALEN(X) = FIELD(X,18,18)                                 PRNTMIO      30    A
PRNTMIO        AFWA(X) = FIELD(X,0,18)                                  PRNTMIO      31    A
PRNTMIO                                                                 PRNTMIO      32    A
PRNTMIO  *         BLOCK HEADER WORD                                    PRNTMIO      33    A
PRNTMIO                                                                 PRNTMIO      34    A
PRNTMIO        HAV(X) = FIELD(X,59,1)                                   PRNTMIO      35    A
PRNTMIO        HPRI(X) = FIELD(X,58,1)                                  PRNTMIO      36    A
PRNTMIO        HHB(X) = FIELD(X,57,1)                                   PRNTMIO      37    A
PRNTMIO        HFW(X) = FIELD(X,56,1)                                   PRNTMIO      38    A
PRNTMIO        HBI(X) = FIELD(X,36,18)                                  PRNTMIO      39    A
PRNTMIO        HBST(X) = FIELD(X,18,18)                                 PRNTMIO      40    A
PRNTMIO        HLEN(X) = FIELD(X,0,18)                                  PRNTMIO      41    A
PRNTMIO                                                                 PRNTMIO      42    A
PRNTMIO  *         BLOCK TRAILER WORD                                   PRNTMIO      43    A
PRNTMIO                                                                 PRNTMIO      44    A
PRNTMIO        TAV(X) = FIELD(X,59,1)                                   PRNTMIO      45    A
PRNTMIO        TLKF(X) = FIELD(X,18,18)                                 PRNTMIO      46    A
PRNTMIO        TLKB(X) = FIELD(X,36,18)                                 PRNTMIO      47    A
PRNTMIO        TLEN(X) = FIELD(X,0,18)                                  PRNTMIO      48    A
PRNTMIO                                                                 PRNTMIO      49    A
PRNTMIO  *         MISC. FIELD DEFS                                     PRNTMIO      50    A
PRNTMIO                                                                 PRNTMIO      51    A
PRNTMIO        ZA(X) = FIELD(X,54,6)                                    PRNTMIO      52    A
PRNTMIO        ZB(X) = FIELD(X,36,18)                                   PRNTMIO      53    A
PRNTMIO        ZC(X) = FIELD(X,18,18)                                   PRNTMIO      54    A
PRNTMIO        ZD(X) = FIELD(X,0,18)                                    PRNTMIO      55    A
PRNTMIO                                                                 PRNTMIO      56    A
PRNTMIO  *         FORMAT SPECIFICATIONS                                PRNTMIO      57    A
PRNTMIO                                                                 PRNTMIO      58    A
PRNTMIO  10    FORMAT(/)                                                PRNTMIO      59    A
PRNTMIO  20    FORMAT(A6)                                               PRNTMIO      60    A
PRNTMIO  30    FORMAT(*--RI WORD--   IC  AT     LEN     OFS             RA     F  PRNTMIO      61    A
PRNTMIO       XWA*)                                                     PRNTMIO      62    A
PRNTMIO  40    FORMAT(*--BA WORD--  BLK PRI  HB  FW      BI     LEN     FWA*)    PRNTMIO      63    A
PRNTMIO  50    FORMAT(*--BH WORD--   AV PRI  HB  FW      BI     BST     LEN*)    PRNTMIO      64    A
PRNTMIO  60    FORMAT(*--BT WORD--   AV         LKF     LKB     LEN*/)          PRNTMIO      65    A
PRNTMIO  70    FORMAT(*--BLOCK INFORMATION TABLE--*/)                   PRNTMIO      66    A
PRNTMIO  80    FORMAT(*--BLOCK STATUS TABLE--*/)                        PRNTMIO      67    A
PRNTMIO  90    FORMAT(*--CONTENTS OF BLK--*/)                           PRNTMIO      68    A
PRNTMIO  100   FORMAT(1H ,7X,Z7,3X)                                     PRNTMIO      69    A
PRNTMIO  110   FORMAT(2I4,2Z8,6X,Z10,Z8)                                PRNTMIO      70    A
PRNTMIO  120   FORMAT(4I4,3Z8)                                          PRNTMIO      71    A
PRNTMIO  130   FORMAT(I4,4X,3Z8)                                        PRNTMIO      72    A
PRNTMIO  140   FORMAT(8X)                                               PRNTMIO      73    A
PRNTMIO  150   FORMAT(10X,Z2,3Z7)                                       PRNTMIO      74    A
PRNTMIO  160   FORMAT(10X)                                              PRNTMIO      75    A
PRNTMIO  165   FORMAT(16X)                                              PRNTMIO      76    A
PRNTMIO  170   FORMAT(10X,*--BIW--*/)                                   PRNTMIO      77    A
PRNTMIO  180   FORMAT(10X,*--BSW--*/)                                   PRNTMIO      78    A
PRNTMIO                                                                 PRNTMIO      79    A
PRNTMIO        LINECM(1) = LINECM(2)                                    PRNTMIO      80    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PRNTMIO

```
PRNTMIO        IF(TEST.EQ.LAB) GO TO 300                                PRNTMIO      81    A
PRNTMIO        TEST = LAB                                               PRNTMIO      82    A
PRNTMIO        PRINT 20, LAB                                            PRNTMIO      83    A
PRNTMIO        GO TO 500                                                PRNTMIO      84    A
PRNTMIO  300   PRINT 20, BLANK                                          PRNTMIO      85    A
PRNTMIO                                                                 PRNTMIO      86    A
PRNTMIO  500   INDXX = INDX.AND.77B                                     PRNTMIO      87    A
PRNTMIO        GO TO(1000,2000,3000,4000,5000,6000,7000),INDXX          PRNTMIO      88    A
PRNTMIO                                                                 PRNTMIO      89    A
PRNTMIO  *         PRINT RANDOM INDEX WORD                              PRNTMIO      90    A
PRNTMIO                                                                 PRNTMIO      91    A
PRNTMIO  1000  CONTINUE                                                 PRNTMIO      92    A
PRNTMIO        PRINT 30                                                 PRNTMIO      93    A
PRNTMIO        PRINT 10                                                 PRNTMIO      94    A
PRNTMIO        MPL = MPW                                                PRNTMIO      95    A
PRNTMIO        ICX = RIC(MPL)                                           PRNTMIO      96    A
PRNTMIO        ATX = RAT(MPL)                                           PRNTMIO      97    A
PRNTMIO        LENX = RLEN(MPL)                                         PRNTMIO      98    A
PRNTMIO        OFSX = ROFS(MPL)                                         PRNTMIO      99    A
PRNTMIO        RAX = RRA(MPL)                                           PRNTMIO     100    A
PRNTMIO        FWAX = RFWA(MPL)                                         PRNTMIO     101    A
PRNTMIO        LCMX = RLCM(MPL)                                         PRNTMIO     102    A
PRNTMIO        IF(ICX.EQ.0) GO TO 1100                                  PRNTMIO     103    A
PRNTMIO        OFSX = 0                                                 PRNTMIO     104    A
PRNTMIO        RAX = LCMX                                               PRNTMIO     105    A
PRNTMIO        GO TO 1200                                               PRNTMIO     106    A
PRNTMIO  1100  FWAX = 0                                                 PRNTMIO     107    A
PRNTMIO        IF(ATX.EQ.1) RAX = LCMX                                  PRNTMIO     108    A
PRNTMIO  1200  PRINT 100,RL                                             PRNTMIO     109    A
PRNTMIO        PRINT 110, ICX,ATX,LENX,OFSX,RAX,FWAX                    PRNTMIO     110    A
PRNTMIO        PRINT 10                                                 PRNTMIO     111    A
PRNTMIO        GO TO 9000                                               PRNTMIO     112    A
PRNTMIO                                                                 PRNTMIO     113    A
PRNTMIO  *         PRINT BLOCK ADDRESS WORD                             PRNTMIO     114    A
PRNTMIO                                                                 PRNTMIO     115    A
PRNTMIO  2000  CONTINUE                                                 PRNTMIO     116    A
PRNTMIO        PRINT 40                                                 PRNTMIO     117    A
PRNTMIO        PRINT 10                                                 PRNTMIO     118    A
PRNTMIO        MPL = MPW                                                PRNTMIO     119    A
PRNTMIO        BLKX = ABLK(MPL)                                         PRNTMIO     120    A
PRNTMIO        PRIX = APRI(MPL)                                         PRNTMIO     121    A
PRNTMIO        HBX = AHB(MPL)                                           PRNTMIO     122    A
PRNTMIO        FWX = AFW(MPL)                                           PRNTMIO     123    A
PRNTMIO        BIX = ABI(MPL)                                           PRNTMIO     124    A
PRNTMIO        LENX = ALEN(MPL)                                         PRNTMIO     125    A
PRNTMIO        FWAX = AFWA(MPL)                                         PRNTMIO     126    A
PRNTMIO        PRINT 100, RL                                            PRNTMIO     127    A
PRNTMIO        PRINT 120, BLKX,PRIX,HBX,FWX,BIX,LENX,FWAX               PRNTMIO     128    A
PRNTMIO        PRINT 10                                                 PRNTMIO     129    A
PRNTMIO        GO TO 9000                                               PRNTMIO     130    A
PRNTMIO                                                                 PRNTMIO     131    A
PRNTMIO  *         PRINT BLOCK HEADER WORD                              PRNTMIO     132    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PRNTMIO

```
PRNTMIO                                                                  PRNTMIO      133     A
PRNTMIO  3000  CONTINUE                                                  PRNTMIO      134     A
PRNTMIO        PRINT 50                                                  PRNTMIO      135     A
PRNTMIO        PRINT 10                                                  PRNTMIO      136     A
PRNTMIO        MPL = MPW                                                 PRNTMIO      137     A
PRNTMIO        AVX = HAV(MPL)                                            PRNTMIO      138     A
PRNTMIO        PRIX = HPRI(MPL)                                          PRNTMIO      139     A
PRNTMIO        HBX = HHB(MPL)                                            PRNTMIO      140     A
PRNTMIO        FWX = HFW(MPL)                                            PRNTMIO      141     A
PRNTMIO        BIX = HBI(MPL)                                            PRNTMIO      142     A
PRNTMIO        BSTX = HBST(MPL)                                          PRNTMIO      143     A
PRNTMIO        LENX = HLEN(MPL)                                          PRNTMIO      144     A
PRNTMIO        PRINT 100, RL                                             PRNTMIO      145     A
PRNTMIO        PRINT 120, AVX,PRIX,HBX,FWX,BIX,BSTX,LENX                 PRNTMIO      146     A
PRNTMIO        PRINT 10                                                  PRNTMIO      147     A
PRNTMIO        GO TO 9000                                                PRNTMIO      148     A
PRNTMIO                                                                  PRNTMIO      149     A
PRNTMIO  *         PRINT BLOCK TRAILER WORD                              PRNTMIO      150     A
PRNTMIO                                                                  PRNTMIO      151     A
PRNTMIO  4000  CONTINUE                                                  PRNTMIO      152     A
PRNTMIO        PRINT 60                                                  PRNTMIO      153     A
PRNTMIO        MPL = MPW                                                 PRNTMIO      154     A
PRNTMIO        AVX = TAV(MPL)                                            PRNTMIO      155     A
PRNTMIO        LKFX = TLKF(MPL)                                          PRNTMIO      156     A
PRNTMIO        LKBX = TLKB(MPL)                                          PRNTMIO      157     A
PRNTMIO        LENX = TLEN(MPL)                                          PRNTMIO      158     A
PRNTMIO        PRINT 100, RL                                             PRNTMIO      159     A
PRNTMIO        PRINT 130, AVX,LKFX,LKBX,LENX                             PRNTMIO      160     A
PRNTMIO        PRINT 10                                                  PRNTMIO      161     A
PRNTMIO        GO TO 9000                                                PRNTMIO      162     A
PRNTMIO                                                                  PRNTMIO      163     A
PRNTMIO  *         PRINT BIT TABLE                                       PRNTMIO      164     A
PRNTMIO                                                                  PRNTMIO      165     A
PRNTMIO  5000  CONTINUE                                                  PRNTMIO      166     A
PRNTMIO        PRINT 10                                                  PRNTMIO      167     A
PRNTMIO        PRINT 20, BLANK                                           PRNTMIO      168     A
PRNTMIO        PRINT 70                                                  PRNTMIO      169     A
PRNTMIO        PRINT 20, BLANK                                           PRNTMIO      170     A
PRNTMIO        PRINT 30                                                  PRNTMIO      171     A
PRNTMIO        PRINT 170                                                 PRNTMIO      172     A
PRNTMIO        DO 5500 I = 1,LBIT,2                                      PRNTMIO      173     A
PRNTMIO        MPZ = OBIT(I)                                             PRNTMIO      174     A
PRNTMIO        ZAX = ZA(MPZ)                                             PRNTMIO      175     A
PRNTMIO        ZBX = ZB(MPZ)                                             PRNTMIO      176     A
PRNTMIO        ZCX = ZC(MPZ)                                             PRNTMIO      177     A
PRNTMIO        ZDX = ZD(MPZ)                                             PRNTMIO      178     A
PRNTMIO        MPL = OBIT(I+1)                                           PRNTMIO      179     A
PRNTMIO        MPLX = I-1                                                PRNTMIO      180     A
PRNTMIO        ICX = RIC(MPL)                                            PRNTMIO      181     A
PRNTMIO        ATX = RAT(MPL)                                            PRNTMIO      182     A
PRNTMIO        LENX = RLEN(MPL)                                          PRNTMIO      183     A
PRNTMIO        OFSX = ROFS(MPL)                                          PRNTMIO      184     A

               0        1         2         3         4         5         6         7         8
               12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PRNTMIO

```
PRNTMIO        RAX = RRA(MPL)                                          PRNTMIO    185    A
PRNTMIO        FWAX = RFWA(MPL)                                        PRNTMIO    186    A
PRNTMIO        LCMX = RLCM(MPL)                                        PRNTMIO    187    A
PRNTMIO        IF(ICX.EQ.0) GO TO 5100                                 PRNTMIO    188    A
PRNTMIO        OFSX = 0                                                PRNTMIO    189    A
PRNTMIO        RAX = LCMX                                              PRNTMIO    190    A
PRNTMIO        GO TO 5200                                              PRNTMIO    191    A
PRNTMIO  5100  FWAX = 0                                                PRNTMIO    192    A
PRNTMIO        IF(ATX.EQ.1) RAX = LCMX                                 PRNTMIO    193    A
PRNTMIO  5200  PRINT 100, MPLX                                         PRNTMIO    194    A
PRNTMIO        PRINT 110, ICX,ATX,LENX,OFSX,RAX,FWAX                   PRNTMIO    195    A
PRNTMIO        PRINT 150, ZAX,ZBX,ZCX,ZDX                              PRNTMIO    196    A
PRNTMIO        PRINT 10                                                PRNTMIO    197    A
PRNTMIO  5500  CONTINUE                                                PRNTMIO    198    A
PRNTMIO        GO TO 9000                                              PRNTMIO    199    A
PRNTMIO                                                                PRNTMIO    200    A
PRNTMIO  *         PRINT BST TABLE                                     PRNTMIO    201    A
PRNTMIO                                                                PRNTMIO    202    A
PRNTMIO  6000  CONTINUE                                                PRNTMIO    203    A
PRNTMIO        PRINT 10                                                PRNTMIO    204    A
PRNTMIO        PRINT 20, BLANK                                         PRNTMIO    205    A
PRNTMIO        PRINT 80                                                PRNTMIO    206    A
PRNTMIO        PRINT 20, BLANK                                         PRNTMIO    207    A
PRNTMIO        PRINT 40                                                PRNTMIO    208    A
PRNTMIO        PRINT 180                                               PRNTMIO    209    A
PRNTMIO        DO 6500 I = 1,LBST,2                                    PRNTMIO    210    A
PRNTMIO        MPL = OBST(I)                                           PRNTMIO    211    A
PRNTMIO        MPLX = I-1                                              PRNTMIO    212    A
PRNTMIO        BLKX = ABLK(MPL)                                        PRNTMIO    213    A
PRNTMIO        PRIX = APRI(MPL)                                        PRNTMIO    214    A
PRNTMIO        HBX = AHB(MPL)                                          PRNTMIO    215    A
PRNTMIO        FWX = AFW(MPL)                                          PRNTMIO    216    A
PRNTMIO        BIX = ABI(MPL)                                          PRNTMIO    217    A
PRNTMIO        LENX = ALEN(MPL)                                        PRNTMIO    218    A
PRNTMIO        FWAX = AFWA(MPL)                                        PRNTMIO    219    A
PRNTMIO        PRINT 100, MPLX                                         PRNTMIO    220    A
PRNTMIO        PRINT 120, BLKX,PRIX,HBX,FWX,BIX,LENX,FWAX              PRNTMIO    221    A
PRNTMIO        MPZ = OBST(I+1)                                         PRNTMIO    222    A
PRNTMIO        ZAX = ZA(MPZ)                                           PRNTMIO    223    A
PRNTMIO        ZBX = ZB(MPZ)                                           PRNTMIO    224    A
PRNTMIO        ZCX = ZC(MPZ)                                           PRNTMIO    225    A
PRNTMIO        ZDX = ZD(MPZ)                                           PRNTMIO    226    A
PRNTMIO        PRINT 140                                               PRNTMIO    227    A
PRNTMIO        PRINT 150, ZAX,ZBX,ZCX,ZDX                              PRNTMIO    228    A
PRNTMIO        PRINT 10                                                PRNTMIO    229    A
PRNTMIO  6500  CONTINUE                                                PRNTMIO    230    A
PRNTMIO        GO TO 9000                                              PRNTMIO    231    A
PRNTMIO                                                                PRNTMIO    232    A
PRNTMIO  *         PRINT CONTENTS OF BLK                               PRNTMIO    233    A
PRNTMIO                                                                PRNTMIO    234    A
PRNTMIO  7000  CONTINUE                                                PRNTMIO    235    A
PRNTMIO        PRINT 10                                                PRNTMIO    236    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PRNTMIO

```
PRNTMIO        PRINT 20, BLANK                                          PRNTMIO    237    A
PRNTMIO        PRINT 90                                                 PRNTMIO    238    A
PRNTMIO        I = 1                                                    PRNTMIO    239    A
PRNTMIO  7100  I = I+1                                                  PRNTMIO    240    A
PRNTMIO        MPL = OBLK(I)                                            PRNTMIO    241    A
PRNTMIO        IF(MPL.EQ.0) GO TO 9000                                  PRNTMIO    242    A
PRNTMIO        PRINT 20, BLANK                                          PRNTMIO    243    A
PRNTMIO        PRINT 50                                                 PRNTMIO    244    A
PRNTMIO        PRINT 165                                                PRNTMIO    245    A
PRNTMIO        PRINT 60                                                 PRNTMIO    246    A
PRNTMIO        MPLX = LOCF(OBLK(I))                                     PRNTMIO    247    A
PRNTMIO        AVX = HAV(MPL)                                           PRNTMIO    248    A
PRNTMIO        PRIX = HPRI(MPL)                                         PRNTMIO    249    A
PRNTMIO        HBX = HHB(MPL)                                           PRNTMIO    250    A
PRNTMIO        FWX = HFW(MPL)                                           PRNTMIO    251    A
PRNTMIO        BIX = HBI(MPL)                                           PRNTMIO    252    A
PRNTMIO        BSTX = HBST(MPL)                                         PRNTMIO    253    A
PRNTMIO        LENX = HLEN(MPL)                                         PRNTMIO    254    A
PRNTMIO        PRINT 100, MPLX                                          PRNTMIO    255    A
PRNTMIO        PRINT 120, AVX,PRIX,HBX,FWX,BIX,BSTX,LENX                PRNTMIO    256    A
PRNTMIO        PRINT 160                                                PRNTMIO    257    A
PRNTMIO        K = I+1                                                  PRNTMIO    258    A
PRNTMIO        I = I+LENX-1                                             PRNTMIO    259    A
PRNTMIO        MPL = OBLK(I)                                            PRNTMIO    260    A
PRNTMIO        MPLX = LOCF(OBLK(I))                                     PRNTMIO    261    A
PRNTMIO        AVX = TAV(MPL)                                           PRNTMIO    262    A
PRNTMIO        LKFX = TLKF(MPL)                                         PRNTMIO    263    A
PRNTMIO        LKBX = TLKB(MPL)                                         PRNTMIO    264    A
PRNTMIO        LENX = TLEN(MPL)                                         PRNTMIO    265    A
PRNTMIO        PRINT 100, MPLX                                          PRNTMIO    266    A
PRNTMIO        PRINT 130, AVX,LKFX,LKBX,LENX                            PRNTMIO    267    A
PRNTMIO        PRINT 10                                                 PRNTMIO    268    A
PRNTMIO        IF(AVX.EQ.1) GO TO 7100                                  PRNTMIO    269    A
PRNTMIO        IF(FWX.EQ.1) GO TO 7100                                  PRNTMIO    270    A
PRNTMIO        IF(NRL.EQ.1) GO TO 7100                                  PRNTMIO    271    A
PRNTMIO        LTXT = LENX-2                                            PRNTMIO    272    A
PRNTMIO        CTR(1) = 0                                               PRNTMIO    273    A
PRNTMIO        CTR(2) = 10HPRNTBLK                                      PRNTMIO    274    A
PRNTMIO        CTR(3) = 1                                               PRNTMIO    275    A
PRNTMIO        CTR(4) = 2                                               PRNTMIO    276    A
PRNTMIO        CTR(5) = 1                                               PRNTMIO    277    A
PRNTMIO        CALL DMPRLST(CTR,OBLK(K),LTXT)                           PRNTMIO    278    A
PRNTMIO        LINECM(1) = LINECM(2)                                    PRNTMIO    279    A
PRNTMIO        GO TO 7100                                               PRNTMIO    280    A
PRNTMIO                                                                 PRNTMIO    281    A
PRNTMIO  *         END PROCESSING                                       PRNTMIO    282    A
PRNTMIO                                                                 PRNTMIO    283    A
PRNTMIO  9000  LINECM(1) = 0                                            PRNTMIO    284    A
PRNTMIO        RETURN                                                   PRNTMIO    285    A
PRNTMIO        END                                                      PRNTMIO    286    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - PRNTMIO

```
IDENTIFIER                TOTAL    ACTIVE

PRNTMIO                     286       286

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PRNTUDI

PRNTUDI  *DECK     PRNTUDI                                                  PRNTUDI        1     A
PRNTUDI            SUBROUTINE PRNTUDI( UDI , N )                            PRNTUDI        2     A
PRNTUDI                                                                     PRNTUDI        3     A
PRNTUDI  *         PRNT UDI - PRINT *USE/DEF* BITS SET IN *UDT* ( GPO / GRA ) PRNTUDI        4     A
PRNTUDI                                                                     PRNTUDI        5     A
PRNTUDI            IMPLICIT INTEGER(A-Z)                                    PRNTUDI        6     A
PRNTUDI            INTEGER UDI(N),BVAL(12),BV(7)                            PRNTUDI        7     A
PRNTUDI            COMMON/LINE/LINECM(2)                                    PRNTUDI        8     A
PRNTUDI                                                                     PRNTUDI        9     A
PRNTUDI            DATA BVAL / 1H  ,5H DEFR , 1H ,4H DEF , 1H  , 4H DBU     PRNTUDI       10     A
PRNTUDI       1  , 1H  , 4H USE , 1H  , 4H ^RD , 1H  , 4H IST /             PRNTUDI       11     A
PRNTUDI                                                                     PRNTUDI       12     A
PRNTUDI            FIELD(X,P,L) = .N.MASK(60-L).A.SHIFT(X,60-P)             PRNTUDI       13     A
PRNTUDI            STC(X) = FIELD(X,45,9)                                   PRNTUDI       14     A
PRNTUDI                                                                     PRNTUDI       15     A
PRNTUDI            PRINT 1                                                  PRNTUDI       16     A
PRNTUDI   1        FORMAT (/*  -- USE/DEF BIT SETTING --*/)                 PRNTUDI       17     A
PRNTUDI   2        FORMAT(/)                                                PRNTUDI       18     A
PRNTUDI                                                                     PRNTUDI       19     A
PRNTUDI            LINECM = LINECM(2)                                       PRNTUDI       20     A
PRNTUDI            PC = 1                                                   PRNTUDI       21     A
PRNTUDI                                                                     PRNTUDI       22     A
PRNTUDI            DO 20 I = 1,N,2                                          PRNTUDI       23     A
PRNTUDI            B = MASK(6) .A. UDI(I)                                   PRNTUDI       24     A
PRNTUDI            IF( B .EQ. 0 ) GO TO 20                                  PRNTUDI       25     A
PRNTUDI                                                                     PRNTUDI       26     A
PRNTUDI            DO 15 K = 1,6                                            PRNTUDI       27     A
PRNTUDI            B = SHIFT(B,1)                                           PRNTUDI       28     A
PRNTUDI            J = B.A.1                                                PRNTUDI       29     A
PRNTUDI   15       BV(K) = BVAL(2*K+J-1)                                    PRNTUDI       30     A
PRNTUDI                                                                     PRNTUDI       31     A
PRNTUDI            BV(7) = STC( UDI(I) )                                    PRNTUDI       32     A
PRNTUDI            J = I/2                                                  PRNTUDI       33     A
PRNTUDI            PRINT 11,J,BV                                            PRNTUDI       34     A
PRNTUDI   11       FORMAT(I8*U *A5,5A4,I4)                                  PRNTUDI       35     A
PRNTUDI                                                                     PRNTUDI       36     A
PRNTUDI            PC = PC + 1                                              PRNTUDI       37     A
PRNTUDI            IF( (PC.A.1) .EQ. 0 ) PRINT 2                            PRNTUDI       38     A
PRNTUDI                                                                     PRNTUDI       39     A
PRNTUDI   20       CONTINUE                                                 PRNTUDI       40     A
PRNTUDI                                                                     PRNTUDI       41     A
PRNTUDI            IF( (PC.A.1) .NE. 0 ) PRINT 2                            PRNTUDI       42     A
PRNTUDI            LINECM = 0                                               PRNTUDI       43     A
PRNTUDI                                                                     PRNTUDI       44     A
PRNTUDI            RETURN                                                   PRNTUDI       45     A
PRNTUDI                                                                     PRNTUDI       46     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PRNTUDI

PRNTUDI                 END                                                                     PRNTUDI        47     A


            SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - PRNTUDI

                IDENTIFIER                  TOTAL    ACTIVE

                PRNTUDI                        47        47

                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PRNTABV

PRNTABV  *DECK          PRNTABV                                                                 PRNTABV         1     A
PRNTABV                 SUBROUTINE PRNTABV( NAME , BVT )                                        PRNTABV         2     A
PRNTABV                 IMPLICIT INTEGER(A-Z)                                                   PRNTABV         3     A
PRNTABV                                                                                         PRNTABV         4     A
PRNTABV  **             PRNTABV - PRINT A BIT VECTOR                                            PRNTABV         5     A
PRNTABV  *                     NAME = 5L NAM   , OR A RIGHT ADJUSTED BINARY NUMBER              PRNTABV         6     A
PRNTABV  *                     BVT - FWA OF BIT VECTOR                                          PRNTABV         7     A
PRNTABV                                                                                         PRNTABV         8     A
PRNTABV                 COMMON/BVINFO/ VL, NVAR /LINE/LINECM(2)                                 PRNTABV         9     A
PRNTABV                 INTEGER BVT(1),B(3)                                                     PRNTABV        10     A
PRNTABV   1             FORMAT(/)                                                               PRNTABV        11     A
PRNTABV                                                                                         PRNTABV        12     A
PRNTABV                 LINECM = LINECM(2)                                                      PRNTABV        13     A
PRNTABV                 VN = 2-60                                                               PRNTABV        14     A
PRNTABV                                                                                         PRNTABV        15     A
PRNTABV                 DO 40 J = 1,VL                                                          PRNTABV        16     A
PRNTABV                 VN = VN + 60                                                            PRNTABV        17     A
PRNTABV                 IF( SHIFT(NAME,-54) .EQ. 0 ) GO TO 5                                    PRNTABV        18     A
PRNTABV                 PRINT 2,NAME                                                            PRNTABV        19     A
PRNTABV   2             FORMAT(A5)                                                              PRNTABV        20     A
PRNTABV                 GO TO 15                                                                PRNTABV        21     A
PRNTABV   5             PRINT 3,NAME                                                            PRNTABV        22     A
PRNTABV   3             FORMAT(I4,1X)                                                           PRNTABV        23     A
PRNTABV                                                                                         PRNTABV        24     A
PRNTABV   15            VM = MIN0( NVAR+1 , VN+59 )                                             PRNTABV        25     A
PRNTABV                 DO 25 I = VN,VM,3                                                       PRNTABV        26     A
PRNTABV   25            PRINT 26,I                                                              PRNTABV        27     A
PRNTABV   26            FORMAT(I4)                                                              PRNTABV        28     A
PRNTABV                 PRINT 4                                                                 PRNTABV        29     A
PRNTABV   4             FORMAT(/7X)                                                             PRNTABV        30     A
PRNTABV                                                                                         PRNTABV        31     A
PRNTABV                 KG = ( (VM-VN)/3 + 1 ) *3                                               PRNTABV        32     A
PRNTABV                 BW = BVT(J)                                                             PRNTABV        33     A
PRNTABV                                                                                         PRNTABV        34     A
PRNTABV                 DO 30 I = 1,KG,3                                                        PRNTABV        35     A
PRNTABV                 SC = 1 - I                                                              PRNTABV        36     A
PRNTABV                 B(1) = SHIFT(BW,SC).A.1                                                 PRNTABV        37     A
PRNTABV                 B(2) = SHIFT(BW,SC-1).A.1                                               PRNTABV        38     A
PRNTABV                 B(3) = SHIFT(BW,SC-2).A.1                                               PRNTABV        39     A
PRNTABV   30            PRINT 22,B                                                              PRNTABV        40     A
PRNTABV   22            FORMAT(1X,3O1)                                                          PRNTABV        41     A
PRNTABV                 PRINT 1                                                                 PRNTABV        42     A

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PRNTABV

PRNTABV                                                                                  PRNTABV       43     A
PRNTABV   40        CONTINUE                                                             PRNTABV       44     A
PRNTABV                                                                                  PRNTABV       45     A
PRNTABV             LINECM = 0                                                           PRNTABV       46     A
PRNTABV             RETURN                                                               PRNTABV       47     A
PRNTABV                                                                                  PRNTABV       48     A
PRNTABV             END                                                                  PRNTABV       49     A

          SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - PRNTABV

                IDENTIFIER                TOTAL    ACTIVE

                PRNTABV                     49        49

                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PRNTBV

PRNTBV    *DECK     PRNTBV                                                               PRNTBV        1     A
PRNTBV              SUBROUTINE PRNTBV( BVT , INDX , NV )                                 PRNTBV        2     A
PRNTBV                                                                                   PRNTBV        3     A
PRNTBV    **        PRNTBV - PRINT BIT VECTORS / LIVE EXIT VECTORS                       PRNTBV        4     A
PRNTBV                                                                                   PRNTBV        5     A
PRNTBV              IMPLICIT INTEGER(A-Z)                                                PRNTBV        6     A
PRNTBV              INTEGER BVT(1),INDX(1) , B(3) , VECTBL(3)                            PRNTBV        7     A
PRNTBV              COMMON/BVINFO/ VL, NVAR /LINE/LINECM(2)                              PRNTBV        8     A
PRNTBV              DATA VECTBL / 5H DEF  , 5H UBD  , 5H USE   /                         PRNTBV        9     A
PRNTBV                                                                                   PRNTBV       10     A
PRNTBV              LINENR(X) = SHIFT(X,-18).A.77777B                                    PRNTBV       11     A
PRNTBV              SETX(X) = X.A.377777B                                                PRNTBV       12     A
PRNTBV     1        FORMAT(/)                                                            PRNTBV       13     A
PRNTBV     2        FORMAT(/*  -- BIT VECTORS --*)                                       PRNTBV       14     A
PRNTBV     3        FORMAT(1X/)                                                          PRNTBV       15     A
PRNTBV              PRINT 2                                                              PRNTBV       16     A
PRNTBV              K = 1                                                                PRNTBV       17     A
PRNTBV              MODE = 0                                                             PRNTBV       18     A
PRNTBV              WI = INDX + 1                                                        PRNTBV       19     A
PRNTBV                                                                                   PRNTBV       20     A
PRNTBV     5        LINECM = LINECM(2)                                                   PRNTBV       21     A
PRNTBV                                                                                   PRNTBV       22     A
PRNTBV    *         LOOP FOR VECTOR                                                      PRNTBV       23     A
PRNTBV              DO 50 KK = 1,NV                                                      PRNTBV       24     A
PRNTBV              BLINE = 0                                                            PRNTBV       25     A
PRNTBV              VN = 2-60                                                            PRNTBV       26     A
PRNTBV                                                                                   PRNTBV       27     A
PRNTBV              DO 40 J = 1,VL                                                       PRNTBV       28     A
PRNTBV              VN = VN + 60                                                         PRNTBV       29     A
PRNTBV              BW = BVT(WI)                                                         PRNTBV       30     A
PRNTBV              IF( BW .NE. 0 ) GO TO 15                                             PRNTBV       31     A
PRNTBV              IF( (MASK(1).A.BW) .EQ. 0 ) GO TO 40                                 PRNTBV       32     A
PRNTBV                                                                                   PRNTBV       33     A
PRNTBV    15        VM = MIN0( NVAR-1 , VN+59 )                                          PRNTBV       34     A
PRNTBV              IF( BLINE .NE. 0 ) GO TO 16                                          PRNTBV       35     A
PRNTBV              PRINT 3                                                              PRNTBV       36     A

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PRNTBV

```
PRNTBV            BLINE = 1                                                PRNTBV    37    A
PRNTBV    16      IF( K .EQ. 0 ) GO TO 20                                  PRNTBV    38    A
PRNTBV            PRINT 21,VECTBL(K)                                       PRNTBV    39    A
PRNTBV    21      FORMAT(A5)                                               PRNTBV    40    A
PRNTBV            GO TO 24                                                 PRNTBV    41    A
PRNTBV    20      I = LINENR( INDX(2*KK+1) )                               PRNTBV    42    A
PRNTBV            PRINT 23,I                                               PRNTBV    43    A
PRNTBV    23      FORMAT(I5)                                               PRNTBV    44    A
PRNTBV                                                                     PRNTBV    45    A
PRNTBV    24      DO 25 I = VN,VM,3                                        PRNTBV    46    A
PRNTBV    25      PRINT 26,I                                               PRNTBV    47    A
PRNTBV    26      FORMAT(I4)                                               PRNTBV    48    A
PRNTBV                                                                     PRNTBV    49    A
PRNTBV            PRINT 4                                                  PRNTBV    50    A
PRNTBV    4       FORMAT(/7X)                                              PRNTBV    51    A
PRNTBV            KG = VM+1 - VN                                           PRNTBV    52    A
PRNTBV                                                                     PRNTBV    53    A
PRNTBV            DO 30 I = 1,KG,3                                         PRNTBV    54    A
PRNTBV            SC = 1 - I                                               PRNTBV    55    A
PRNTBV            B(1) = SHIFT(BW,SC).A.1                                  PRNTBV    56    A
PRNTBV            B(2) = SHIFT(BW,SC-1).A.1                                PRNTBV    57    A
PRNTBV            B(3) = SHIFT(BW,SC-2).A.1                                PRNTBV    58    A
PRNTBV    30      PRINT 22,B                                               PRNTBV    59    A
PRNTBV    22      FORMAT(1X,3O1)                                           PRNTBV    60    A
PRNTBV            PRINT 1                                                  PRNTBV    61    A
PRNTBV                                                                     PRNTBV    62    A
PRNTBV    40      WI = WI + 1                                              PRNTBV    63    A
PRNTBV            IF( MODE .NE. 0 ) GO TO 45                               PRNTBV    64    A
PRNTBV            K = K + 1                                                PRNTBV    65    A
PRNTBV            GO TO 50                                                 PRNTBV    66    A
PRNTBV    45      WI = SETX( INDX(2*KK+3) ) + MODE                         PRNTBV    67    A
PRNTBV    50      CONTINUE                                                 PRNTBV    68    A
PRNTBV                                                                     PRNTBV    69    A
PRNTBV            LINECM = 0                                               PRNTBV    70    A
PRNTBV            IF( MODE .NE. 0 ) CALL SETST( 0 )                        PRNTBV    71    A
PRNTBV            RETURN                                                   PRNTBV    72    A
PRNTBV                                                                     PRNTBV    73    A
PRNTBV    *       PRNTLX - PRINT PROGRAM LIVE EXIT BIT VECTORS             PRNTBV    74    A
PRNTBV                                                                     PRNTBV    75    A
PRNTBV            ENTRY  PRNTLX                                            PRNTBV    76    A
PRNTBV            CALL SETST( 3 , 28L LINE   LIVE EXIT VECTORS           )  PRNTBV    77    A
PRNTBV            K = 0                                                    PRNTBV    78    A
PRNTBV            MODE = 3*VL + 1                                          PRNTBV    79    A
PRNTBV            WI = SETX( INDX(3) ) + MODE                              PRNTBV    80    A
PRNTBV            GO TO 5                                                  PRNTBV    81    A
PRNTBV                                                                     PRNTBV    82    A
PRNTBV            END                                                      PRNTBV    83    A
```

SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - PRNTBV

```
IDENTIFIER                TOTAL    ACTIVE

0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PRNTBV

                PRNTBV                      83        83

                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PRNTRLI

      PRNTRLI   *DECK     PRNTRLI                                                  PRNTRLI        1     A
      PRNTRLI             SUBROUTINE PRNT RLI( FRW )                               PRNTRLI        2     A
      PRNTRLI             IMPLICIT INTEGER(A-Z)                                    PRNTRLI        3     A
      PRNTRLI                                                                      PRNTRLI        4     A
      PRNTRLI   **        PRNT RLI - PRINT 1 *IL* ( RLIST ) INSTRUCTION            PRNTRLI        5     A
      PRNTRLI                                                                      PRNTRLI        6     A
      PRNTRLI             INTEGER FRW(3),RLI(3),RTAB(3)                            PRNTRLI        7     A
      PRNTRLI             INTEGER JPX(8),JPBB(8)                                   PRNTRLI        8     A
      PRNTRLI             COMMON/OPRS/ OPRS(1)                                     PRNTRLI        9     A
      PRNTRLI             EQUIVALENCE (RL2,RLI(2) )                                PRNTRLI       10     A
      PRNTRLI                                                                      PRNTRLI       11     A
      PRNTRLI             DATA RTAB / 2RB0 , 2RA0 , 2RX0 /                         PRNTRLI       12     A
      PRNTRLI             DATA JPX / 2HZR , 2HNZ , 2HPL , 2HMI , 2HOR , 2HIR , 2HDF PRNTRLI       13           I
                -CCG9
      PRNTRLI             DATA JPX / 2HZR , 2HNZ , 2HPL , 2HMI , 2HIR , 2HOR , 2HDF CCG9         129     A
      PRNTRLI       .           , 2HID /                                           PRNTRLI       14     A
      PRNTRLI             DATA JPBB / 2HEQ , 2HNE , 2HGE , 2HLT /                  PRNTRLI       15     A
      PRNTRLI                                                                      PRNTRLI       16     A
      PRNTRLI             FIELD(X,P,L) = SHIFT(X,60-P).A. .N.MASK(60-L)            PRNTRLI       17     A
      PRNTRLI             KS(X) = FIELD(X,31,1)                                    PRNTRLI       18     A
      PRNTRLI   *                                R1 WORD                           PRNTRLI       19     A
      PRNTRLI             OC(X) = FIELD(X,48,7)                                    PRNTRLI       20     A
      PRNTRLI             RJ(X) = FIELD(X,32,16)                                   PRNTRLI       21     A
      PRNTRLI             RK(X) = FIELD(X,16,16)                                   PRNTRLI       22     A
      PRNTRLI             RI(X) = FIELD(X,0,16)                                    PRNTRLI       23     A
      PRNTRLI   *                                R1 WORD / TYPE II AND III         PRNTRLI       24     A
      PRNTRLI             IN(X) = FIELD(X,30,18)                                   PRNTRLI       25     A
      PRNTRLI             SO(X) = FIELD(X,16,13)                                   PRNTRLI       26     A
      PRNTRLI             H2(X) = FIELD(X,18,12)                                   PRNTRLI       27     A
      PRNTRLI   *                                R2 OR IH WORD                     PRNTRLI       28     A
      PRNTRLI             RF(X) = FIELD(X,36,18)                                   PRNTRLI       29     A
      PRNTRLI             CA(X) = FIELD(X,18,18)                                   PRNTRLI       30     A
      PRNTRLI             IH(X) = FIELD(X,0,18)                                    PRNTRLI       31     A
      PRNTRLI                                                                      PRNTRLI       32     A
      PRNTRLI             RLI(1) = IABS( FRW(1) )                                  PRNTRLI       33     A
      PRNTRLI             RL2 = FRW(2)                                             PRNTRLI       34     A
      PRNTRLI             RLI(3) = FRW(3)                                          PRNTRLI       35     A
      PRNTRLI             OCV = OC(RLI)+1                                          PRNTRLI       36     A
      PRNTRLI             RIV = RI(RLI)                                            PRNTRLI       37     A
      PRNTRLI             NAM = OPRS(OCV).A.MASK(24)                               PRNTRLI       38     A
      PRNTRLI             GO TO ( 20 , 30 , 33 , 50 ) OPRS(OCV) .A. 17B            PRNTRLI       39     A
      PRNTRLI                                                                      PRNTRLI       40     A
      PRNTRLI   **        TYPE I INSTRUCTION                                       PRNTRLI       41     A
      PRNTRLI                                                                      PRNTRLI       42     A
      PRNTRLI    20       RJV = RJ(RLI)                                            PRNTRLI       43     A
      PRNTRLI             RKV = RK(RLI)                                            PRNTRLI       44     A
      PRNTRLI             PRINT 21,NAM,RIV,RJV                                     PRNTRLI       45     A
      PRNTRLI    21       FORMAT(2X,A4,Z6,Z7)                                      PRNTRLI       46     A

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PRNTRLI

```
PRNTRLI            IF( RKV .EQ. 0 ) GO TO 23                                PRNTRLI     47    A
PRNTRLI            PRINT 22,RKV                                             PRNTRLI     48    A
PRNTRLI    22      FORMAT(Z7)                                               PRNTRLI     49    A
PRNTRLI            GO TO 25                                                 PRNTRLI     50    A
PRNTRLI                                                                     PRNTRLI     51    A
PRNTRLI    23      PRINT 24                                                 PRNTRLI     52    A
PRNTRLI    24      FORMAT(7X)                                               PRNTRLI     53    A
PRNTRLI                                                                     PRNTRLI     54    A
PRNTRLI    25      IF( RL2 .EQ. 0 ) GO TO 100                               PRNTRLI     55    A
PRNTRLI            CAV = CA(RL2)                                            PRNTRLI     56    A
PRNTRLI            RFV = RF(RL2)                                            PRNTRLI     57    A
PRNTRLI            H2V = 0                                                  PRNTRLI     58    A
PRNTRLI            GO TO 39                                                 PRNTRLI     59    A
PRNTRLI                                                                     PRNTRLI     60    A
PRNTRLI  **        TYPE II INSTRUCTION                                      PRNTRLI     61    A
PRNTRLI                                                                     PRNTRLI     62    A
PRNTRLI    30      INV = IN(RLI)                                            PRNTRLI     63    A
PRNTRLI            PRINT 31,NAM,RIV,INV                                     PRNTRLI     64    A
PRNTRLI    31      FORMAT(2X,A4,Z6,Z7,*B*)                                  PRNTRLI     65    A
PRNTRLI                                                                     PRNTRLI     66    A
PRNTRLI            SOX = SO(RLI)                                            PRNTRLI     67    A
PRNTRLI            IF( SOX. EQ. 0 ) GO TO 100                               PRNTRLI     68    A
PRNTRLI            INV = FIELD(SOX,8,2)                                     PRNTRLI     69    A
PRNTRLI                              SOX = SHIFT( 2R0  + (300B.A.SOX) , 12 ) .O.  PRNTRLI     70    A
PRNTRLI       .          ( RTAB( (SOX.A.70B)/8 + 1 ) +(SOX.A.7B) )          PRNTRLI     71    A
PRNTRLI            PRINT 32,INV,SOX                                         PRNTRLI     72    A
PRNTRLI    32      FORMAT(Z3,1X,R4,4X)                                      PRNTRLI     73    A
PRNTRLI                                                                     PRNTRLI     74    A
PRNTRLI            IF( RL2 .EQ. 0 ) GO TO 100                               PRNTRLI     75    A
PRNTRLI            RFV = RF(RL2)                                            PRNTRLI     76    A
PRNTRLI            CAV = CA(RL2)                                            PRNTRLI     77    A
PRNTRLI            H2V = 0                                                  PRNTRLI     78    A
PRNTRLI            PRINT 26                                                 PRNTRLI     79    A
PRNTRLI    26      FORMAT(10X)                                              PRNTRLI     80    A
PRNTRLI            GO TO 40                                                 PRNTRLI     81    A
PRNTRLI                                                                     PRNTRLI     82    A
PRNTRLI  **        TYPE III INSTRUCTION                                     PRNTRLI     83    A
PRNTRLI                                                                     PRNTRLI     84    A
PRNTRLI    33      CAV = CA(RL2)                                            PRNTRLI     85    A
PRNTRLI            RFV = RF(RL2)                                            PRNTRLI     86    A
PRNTRLI            IF( (NAM.A.MASK(12)) .NE. 2LJP ) GO TO 35                PRNTRLI     87    A
PRNTRLI            I = NAM                                                  PRNTRLI     88    A
PRNTRLI            NAM = JPX( (CAV.AND.63)+1 )                              PRNTRLI     89    A
PRNTRLI            IF( I .EQ. 4LJPX  ) GO TO 34                             PRNTRLI     90    A
PRNTRLI            NAM = JPBB( (CAV.AND.63)+1 )                             PRNTRLI     91    A
PRNTRLI    34      CAV = SHIFT(CAV,-6)                                      PRNTRLI     92    A
PRNTRLI            GO TO 37                                                 PRNTRLI     93    A
PRNTRLI                                                                     PRNTRLI     94    A
PRNTRLI    35      IF( NAM .NE. 4LRJXJ ) GO TO 37                           PRNTRLI     95    A
PRNTRLI            NAM = 2LRJ.OR.SHIFT(JPX( (CAV.A.63)+1 ) , -12 )          PRNTRLI     96    A
PRNTRLI            CAV = SHIFT(CAV,-6)                                      PRNTRLI     97    A
PRNTRLI                                                                     PRNTRLI     98    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PRNTRLI

```
PRNTRLI   37       PRINT 21,NAM,RIV,RFV                                  PRNTRLI      99     A
PRNTRLI                                                                  PRNTRLI     100     A
PRNTRLI            IF( KS(RLI(3)) .EQ. 0 ) GO TO 38                      PRNTRLI     101     A
PRNTRLI            PRINT 135,CAV                                         PRNTRLI     102     A
PRNTRLI   135      FORMAT(I6)                                            PRNTRLI     103     A
PRNTRLI            GO TO 100                                             PRNTRLI     104     A
PRNTRLI                                                                  PRNTRLI     105     A
PRNTRLI   38       PRINT 24                                              PRNTRLI     106     A
PRNTRLI            H2V = H2(RLI)                                         PRNTRLI     107     A
PRNTRLI            INV = IN(RLI)/2                                       PRNTRLI     108     A
PRNTRLI            IF( INV .EQ. 0 ) GO TO 39                             PRNTRLI     109     A
PRNTRLI            PRINT 138,INV                                         PRNTRLI     110     A
PRNTRLI   138      FORMAT(I5*U*)                                         PRNTRLI     111     A
PRNTRLI            GO TO 40                                              PRNTRLI     112     A
PRNTRLI                                                                  PRNTRLI     113     A
PRNTRLI   39       PRINT 139                                             PRNTRLI     114     A
PRNTRLI   139      FORMAT(6X)                                            PRNTRLI     115     A
PRNTRLI                                                                  PRNTRLI     116     A
PRNTRLI  *         PRINT *IH* WORD - * IH +CA  RF -H2 *                  PRNTRLI     117     A
PRNTRLI   40       IHV = IH(RL2)                                         PRNTRLI     118     A
PRNTRLI            IF( RFV .GE. 4 ) GO TO 41                             PRNTRLI     119         I
          -CCG9
PRNTRLI            IF( RFV .NE. 1 ) GO TO 41                             CCG9        130     A
PRNTRLI            SYV = 8L -A0-                                         PRNTRLI     120     A
PRNTRLI            GO TO 42                                              PRNTRLI     121     A
PRNTRLI   41       SYV = SYM(IHV)                                        PRNTRLI     122     A
PRNTRLI                                                                  PRNTRLI     123     A
PRNTRLI   42       PRINT 140,SYV                                         PRNTRLI     124     A
PRNTRLI   140      FORMAT(1X,A8)                                         PRNTRLI     125     A
PRNTRLI                                                                  PRNTRLI     126     A
PRNTRLI            IF( (CAV+RFV/2) .EQ. 0 ) GO TO 45                     PRNTRLI     127     A
PRNTRLI            PRINT 142,CAV                                         PRNTRLI     128     A
PRNTRLI   142      FORMAT(*+*O6)                                         PRNTRLI     129     A
PRNTRLI                                                                  PRNTRLI     130     A
PRNTRLI            IF( RFV/2 .EQ. 0 ) GO TO 45                           PRNTRLI     131     A
PRNTRLI            PRINT 143,RFV                                         PRNTRLI     132     A
PRNTRLI   143      FORMAT(*  R*O6)                                       PRNTRLI     133     A
PRNTRLI                                                                  PRNTRLI     134     A
PRNTRLI   45       IF( H2V .EQ. 0 ) GO TO 100                            PRNTRLI     135     A
PRNTRLI            SYV = SYM(H2V)                                        PRNTRLI     136     A
PRNTRLI            PRINT 145,SYV                                         PRNTRLI     137     A
PRNTRLI   145      FORMAT(*  -*A7)                                       PRNTRLI     138     A
PRNTRLI            GO TO 100                                             PRNTRLI     139     A
PRNTRLI                                                                  PRNTRLI     140     A
PRNTRLI  **        TYPE IV INSTRUCTION                                   PRNTRLI     141     A
PRNTRLI                                                                  PRNTRLI     142     A
PRNTRLI   50       CAV = IN(RLI)                                         PRNTRLI     143     A
PRNTRLI            IHV = IH(RLI)                                         PRNTRLI     144     A
PRNTRLI            INV = H2(RLI)                                         PRNTRLI     145     A
PRNTRLI            IF( OCV .NE. 2 ) GO TO 52                             PRNTRLI     146     A
PRNTRLI  *                                   BOS                         PRNTRLI     147     A
PRNTRLI            PRINT 51,CAV,IHV,INV                                  PRNTRLI     148     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
              LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PRNTRLI

PRNTRLI   51       FORMAT(* STMT*I5,Z6*B*Z2)                                    PRNTRLI      149     A
PRNTRLI            GO TO 60                                                     PRNTRLI      150     A
PRNTRLI                                                                          PRNTRLI      151     A
PRNTRLI   52       IF( OCV .NE. 3 ) GO TO 54                                    PRNTRLI      152     A
PRNTRLI  *                                       EOS                             PRNTRLI      153     A
PRNTRLI            PRINT 53,NAM,IHV,INV                                         PRNTRLI      154     A
PRNTRLI   53       FORMAT(2X,A4,Z10*B*Z2)                                       PRNTRLI      155     A
PRNTRLI            GO TO 60                                                     PRNTRLI      156     A
PRNTRLI                                                                          PRNTRLI      157     A
PRNTRLI  *                                LAB,UJP,RJX,EOQ, ETC                   PRNTRLI      158     A
PRNTRLI   54       SYV = SYM(IHV)                                               PRNTRLI      159     A
PRNTRLI            PRINT 57,NAM,SYV,CAV                                         PRNTRLI      160     A
PRNTRLI   57       FORMAT(2X,A4,3X,A7,* / *O6)                                  PRNTRLI      161     A
PRNTRLI                                                                          PRNTRLI      162     A
PRNTRLI   60       IF( RL2 .EQ. 0 ) GO TO 100                                   PRNTRLI      163     A
PRNTRLI            RIV = FIELD(RL2,0,18)                                        PRNTRLI      164     A
PRNTRLI            RJV = FIELD(RL2,18,18)                                       PRNTRLI      165     A
PRNTRLI            RKV = FIELD(RL2,36,18)                                       PRNTRLI      166     A
PRNTRLI            OCV = FIELD(RL2,54,6)                                        PRNTRLI      167     A
PRNTRLI            PRINT 61,OCV,RKV,RJV,RIV                                     PRNTRLI      168     A
PRNTRLI   61       FORMAT(4X,O2,3Z6)                                            PRNTRLI      169     A
PRNTRLI                                                                          PRNTRLI      170     A
PRNTRLI   100      PRINT 101                                                    PRNTRLI      171     A
PRNTRLI   101      FORMAT(/)                                                    PRNTRLI      172     A
PRNTRLI            RETURN                                                       PRNTRLI      173     A
PRNTRLI                                                                          PRNTRLI      174     A
PRNTRLI            END                                                          PRNTRLI      175     A

          SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - PRNTRLI

              IDENTIFIER                TOTAL    ACTIVE

              PRNTRLI                     175       173
              CCG9                          2         2

              LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PRNTMCG

PRNTMCG  *DECK     PRNTMCG                                                       PRNTMCG        1     A
PRNTMCG            SUBROUTINE PRNT MCG( TXT , INDX , LAB )                       PRNTMCG        2     A
PRNTMCG                                                                          PRNTMCG        3     A
PRNTMCG  **        PRNT MCG - *MCG* DEBUGGING PRINT ROUTINES                     PRNTMCG        4     A
PRNTMCG                                                                          PRNTMCG        5     A
PRNTMCG            IMPLICIT INTEGER(A-Z)                                         PRNTMCG        6     A
PRNTMCG            INTEGER TXT(1)                                                PRNTMCG        7     A
PRNTMCG            COMMON/MCG/NINST,NSTORES,SEQTIME,ICL,RMR,DUMMY,PBI(2),DT,IWT  PRNTMCG        8     A
PRNTMCG            COMMON/MCG/AXR,ALR,STL,RVT(24),DUM2,TRA(24),ARA(8),TFU(12)    PRNTMCG        9     A
PRNTMCG            COMMON/MCG/LST(2),LXR,NGX,T,PC,TRS                            PRNTMCG       10     A
PRNTMCG            COMMON/MCG/ JLR,TLAB,TJP,EOQA,IIP(2),EIW,RBV(24)              PRNTMCG       11     A
PRNTMCG                                                                          PRNTMCG       12     A
PRNTMCG            COMMON/LINE/LINECM(2)                                         PRNTMCG       13     A
PRNTMCG            INTEGER REGS(24),FUNITS(2,9),XREGT(2,8)                       PRNTMCG       14     A
PRNTMCG            INTEGER SUBTTL(6)                                             PRNTMCG       15     A

              0         1         2         3         4         5         6         7         8
              12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PRNTMCG

```
PRNTMCG            DATA SUBTTL(1) /                                                  PRNTMCG      16    A
PRNTMCG       .58L  PC  LATE     T   TIS   TEX   TRA   TFU  USES  REG  INST     /    PRNTMCG      17    A
PRNTMCG            DATA REGS / 2RB0,2RB1,2RB2,2RB3,2RB4,2RB5,2RB6,2RB7,              PRNTMCG      18    A
PRNTMCG       .    2RA0,2RA1,2RA2,2RA3,2RA4,2RA5,2RA6,2RA7,2RX0,2RX1,2RX2,2RX3,      PRNTMCG      19    A
PRNTMCG       .    2RX4,2RX5,2RX6,2RX7    /                                          PRNTMCG      20    A
PRNTMCG            DATA (FUNITS(1,I),I=1,9) /5L BOOL , 5L SHFT , 5L IADD ,           PRNTMCG      21    A
PRNTMCG       .    5L FADD , 5L MULT , 5L MUL2 , 5L DIV  , 5L INC1 , 5L INC2  /      PRNTMCG      22    A
PRNTMCG                                                                              PRNTMCG      23    A
PRNTMCG            FIELD(X,P,L) = SHIFT(X,60-P).A. .N.MASK(60-L)                     PRNTMCG      24    A
PRNTMCG                                                                              PRNTMCG      25    A
PRNTMCG            LINK(X) = FIELD(X,0,18)                                           PRNTMCG      26    A
PRNTMCG            RANK(X) = FIELD(X,18,15)                                          PRNTMCG      27    A
PRNTMCG                                                                              PRNTMCG      28    A
PRNTMCG            OC(X) = FIELD(X,48,7)                                             PRNTMCG      29    A
PRNTMCG            RI(X) = FIELD(X,0,16)                                             PRNTMCG      30    A
PRNTMCG            TI(X) = FIELD(X,0,18)                                             PRNTMCG      31    A
PRNTMCG            TE(X) = FIELD(X,18,18)                                            PRNTMCG      32    A
PRNTMCG            REG(X) = FIELD(X,48,6)                                            PRNTMCG      33    A
PRNTMCG            FU(X)  = FIELD(X,48,6)                                            PRNTMCG      34    A
PRNTMCG            LI(X) = FIELD(X,0,18)                                             PRNTMCG      35    A
PRNTMCG                                                                              PRNTMCG      36    A
PRNTMCG            USES(X) = FIELD(X,48,9)                                           PRNTMCG      37    A
PRNTMCG            FT(X) = FIELD(X,53,5)                                             PRNTMCG      38    A
PRNTMCG            SZ(X) = FIELD(X,23,2)                                             PRNTMCG      39    A
PRNTMCG                                                                              PRNTMCG      40    A
PRNTMCG            GO TO ( 1 , 100 , 200 , 300 , 400 , 500 ) INDX                    PRNTMCG      41    A
PRNTMCG                                                                              PRNTMCG      42    A
PRNTMCG  **        PRINT OUT THE ISSUE CANIDATE LIST                                 PRNTMCG      43    A
PRNTMCG                                                                              PRNTMCG      44    A
PRNTMCG   1        IF( ICL .NE. 0 ) GO TO 10                                         PRNTMCG      45    A
PRNTMCG            PRINT 5,LAB                                                       PRNTMCG      46    A
PRNTMCG   5        FORMAT(A6," ** ERROR **, ICL = 0"/)                               PRNTMCG      47    A
PRNTMCG            EOQA = -EOQA                                                      PRNTMCG      48    A
PRNTMCG            RETURN                                                            PRNTMCG      49    A
PRNTMCG                                                                              PRNTMCG      50    A
PRNTMCG   10       LINECM = LINECM(2)                                                PRNTMCG      51    A
PRNTMCG            PRINT 11,LAB,ICL                                                  PRNTMCG      52    A
PRNTMCG   11       FORMAT(/A6* ICL =*Z6)                                             PRNTMCG      53    A
PRNTMCG            J = 1                                                             PRNTMCG      54    A
PRNTMCG  *         RL,RT,RR = RANK LAST, RANK THIS, RANK REVERSAL FLAG (*ERROR*)     PRNTMCG      55    A
PRNTMCG            RL = RR = 0                                                       PRNTMCG      56    A
PRNTMCG            K = ICL                                                           PRNTMCG      57    A
PRNTMCG   20       K = LINK( TXT(K+4) )                                              PRNTMCG      58    A
PRNTMCG            IF( K .EQ. 0 ) GO TO 30                                           PRNTMCG      59    A
PRNTMCG            RT = RANK( TXT(K+4) )                                             PRNTMCG      60    A
PRNTMCG            IF( RT .LT. RL ) RR = K                                           PRNTMCG      61    A
PRNTMCG            RL = RT                                                           PRNTMCG      62    A
PRNTMCG            PRINT 21,K                                                        PRNTMCG      63    A
PRNTMCG   21       FORMAT(Z6)                                                        PRNTMCG      64    A
PRNTMCG            J = J + 1                                                         PRNTMCG      65    A
PRNTMCG            IF( J .LT. 15 ) GO TO 20                                          PRNTMCG      66    A
PRNTMCG            J = 0                                                             PRNTMCG      67    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PRNTMCG

```
PRNTMCG            PRINT 22                                                    PRNTMCG    68    A
PRNTMCG   22       FORMAT(/12X)                                                PRNTMCG    69    A
PRNTMCG            IF( RR .EQ. 0 ) GO TO 20                                    PRNTMCG    70    A
PRNTMCG                                                                        PRNTMCG    71    A
PRNTMCG   30       IF( J .NE. 0 ) PRINT 31                                     PRNTMCG    72    A
PRNTMCG   31       FORMAT(/)                                                   PRNTMCG    73    A
PRNTMCG            IF( RR .EQ. 0 ) GO TO 40                                    PRNTMCG    74    A
PRNTMCG            PRINT 32,RR                                                 PRNTMCG    75    A
PRNTMCG   32       FORMAT(6X,O6,* IS IN WRONG ORDER, RANK REVERSAL OR LOOP IN ICL PRNTMCG 76    A
PRNTMCG       .*/)                                                             PRNTMCG    77    A
PRNTMCG            EOQA = -EOQA                                                PRNTMCG    78    A
PRNTMCG            NRE = NRE + 1                                               PRNTMCG    79    A
PRNTMCG            IF( NRE .LE. 5 ) CALL REMARK(28L -RANK REVERSAL ERROR-       )PRNTMCG  80    A
PRNTMCG                                                                        PRNTMCG    81    A
PRNTMCG   40       LINECM = 0                                                  PRNTMCG    82    A
PRNTMCG            RETURN                                                      PRNTMCG    83    A
PRNTMCG                                                                        PRNTMCG    84    A
PRNTMCG  **        PRINT THE ISSUED INSTRUCTION                                PRNTMCG    85    A
PRNTMCG                                                                        PRNTMCG    86    A
PRNTMCG   100      LIX = LI( PBI(2) )                                          PRNTMCG    87    A
PRNTMCG            LIX = SHIFT( SHIFT(LIX,42) , -42)                           PRNTMCG    88    A
PRNTMCG                                                                        PRNTMCG    89    A
PRNTMCG   101      OCV = OC( TXT(LIX+1) )                                      PRNTMCG    90    A
PRNTMCG            U = USES( TXT(LIX+4) )                                      PRNTMCG    91    A
PRNTMCG            OCV = OC( TXT(LIX+1) )                                      PRNTMCG    92    A
PRNTMCG            IWD = 0                                                     PRNTMCG    93    A
PRNTMCG            LINECM = LINECM(2)                                          PRNTMCG    94    A
PRNTMCG            IF( OCV .GT. 7 ) GO TO 110                                  PRNTMCG    95    A
PRNTMCG  *                          PRINT OUT TIME FOR PSEUDO INSTRUCTION      PRNTMCG    96    A
PRNTMCG            IF( HEADER .NE. 0  ) GO TO 104                              PRNTMCG    97    A
PRNTMCG            PRINT 111,PC,IWD,T                                          PRNTMCG    98    A
PRNTMCG            PRINT 102                                                   PRNTMCG    99    A
PRNTMCG   102      FORMAT(29X)                                                 PRNTMCG   100    A
PRNTMCG            GO TO 120                                                   PRNTMCG   101    A
PRNTMCG                                                                        PRNTMCG   102    A
PRNTMCG   104      PRINT 103,PC,T,U                                            PRNTMCG   103    A
PRNTMCG   103      FORMAT(I4,6X,I6,24X,I6,5X)                                  PRNTMCG   104    A
PRNTMCG            GO TO 120                                                   PRNTMCG   105    A
PRNTMCG                                                                        PRNTMCG   106    A
PRNTMCG   110      TIS = TI(PBI)                                               PRNTMCG   107    A
PRNTMCG            TEX = TE(PBI)                                               PRNTMCG   108    A
PRNTMCG            FUO = FU(PBI(2))                                            PRNTMCG   109    A
PRNTMCG            REGNO = REG(PBI)                                            PRNTMCG   110    A
PRNTMCG            TRAT = TRA(REGNO+1)                                         PRNTMCG   111    A
PRNTMCG            TFUA = TFU(FUO+1)                                           PRNTMCG   112    A
PRNTMCG            REGNO = REGS(REGNO+1)                                       PRNTMCG   113    A
PRNTMCG            IWD = TIS - RANK( TXT(LIX+4) )                              PRNTMCG   114    A
PRNTMCG            MAXEST = MAX0( MAXEST , RANK(TXT(LIX+4))-T )                PRNTMCG   115    A
PRNTMCG                                                                        PRNTMCG   116    A
PRNTMCG            IF( HEADER. EQ. 0 ) GO TO 115                               PRNTMCG   117    A
PRNTMCG            PRINT 114,PC,IWD,T,TIS,TEX,TRAT,TFUA,U,REGNO                PRNTMCG   118    A
PRNTMCG   114      FORMAT(I4,7I6,3X,R2)                                        PRNTMCG   119    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PRNTMCG

```
PRNTMCG            GO TO 120                                                    PRNTMCG      120     A
PRNTMCG                                                                         PRNTMCG      121     A
PRNTMCG   115      PRINT 111,PC,IWD,T,TIS,TEX,TRAT,TFUA,U,REGNO                 PRNTMCG      122     A
PRNTMCG   111      FORMAT(I4,* T,TI,TE,TRA,TFU,USES *7I6,3X,R2)                 PRNTMCG      123     A
PRNTMCG                                                                         PRNTMCG      124     A
PRNTMCG   120      CALL PRNTRLI( TXT( LIX+1 ) )                                 PRNTMCG      125     A
PRNTMCG            LINECM = 0                                                   PRNTMCG      126     A
PRNTMCG   C        CHECK FOR XMT/SA WITH RI = 0 AND SKIP CHECK IF SO            PRNTMCG      127     A
PRNTMCG            IF( RI( TXT(LIX+1) ) .NE. 0 ) GO TO 240                      PRNTMCG      128     A
PRNTMCG            IF( OCV .EQ. 10B ) RETURN                                    PRNTMCG      129     A
PRNTMCG            IF( OCV .NE. 60B ) GO TO 240                                 PRNTMCG      130     A
PRNTMCG            RETURN                                                       PRNTMCG      131     A
PRNTMCG                                                                         PRNTMCG      132     A
PRNTMCG   **       PRINT OUT REGISTER USEAGE INFO                               PRNTMCG      133     A
PRNTMCG   200      CONTINUE                                                     PRNTMCG      134     A
PRNTMCG            PRINT 201,REGS                                               PRNTMCG      135     A
PRNTMCG   201      FORMAT(4X,24(3X,R2))                                         PRNTMCG      136     A
PRNTMCG            LINECM = LINECM(2)                                           PRNTMCG      137     A
PRNTMCG                                                                         PRNTMCG      138     A
PRNTMCG   *        PRINT R-NUMBERS IN REGISTERS                                 PRNTMCG      139     A
PRNTMCG                                                                         PRNTMCG      140     A
PRNTMCG            PRINT 202                                                    PRNTMCG      141     A
PRNTMCG   202      FORMAT(*  RI*)                                               PRNTMCG      142     A
PRNTMCG            DO 220 I = 1,24                                              PRNTMCG      143     A
PRNTMCG            IF( RVT(I) .NE. 0 ) GO TO 215                                PRNTMCG      144     A
PRNTMCG            PRINT 211                                                    PRNTMCG      145     A
PRNTMCG   211      FORMAT(5X)                                                   PRNTMCG      146     A
PRNTMCG            GO TO 220                                                    PRNTMCG      147     A
PRNTMCG   215      PRINT 216,RVT(I)                                             PRNTMCG      148     A
PRNTMCG   216      FORMAT(Z5)                                                   PRNTMCG      149     A
PRNTMCG   220      CONTINUE                                                     PRNTMCG      150     A
PRNTMCG                                                                         PRNTMCG      151     A
PRNTMCG   *        PRINT REMAINING USES OF RI IN REG                            PRNTMCG      152     A
PRNTMCG                                                                         PRNTMCG      153     A
PRNTMCG            PRINT 221                                                    PRNTMCG      154     A
PRNTMCG   221      FORMAT(/*  USES D *)                                         PRNTMCG      155     A
PRNTMCG            DO 230 I = 2,24                                              PRNTMCG      156     A
PRNTMCG            IF( RVT(I) .NE. 0 ) GO TO 225                                PRNTMCG      157     A
PRNTMCG            PRINT 222                                                    PRNTMCG      158     A
PRNTMCG   222      FORMAT(5X)                                                   PRNTMCG      159     A
PRNTMCG            GO TO 230                                                    PRNTMCG      160     A
PRNTMCG                                                                         PRNTMCG      161     A
PRNTMCG   225      U = USES( TXT( RVT(I) + 4 ) )                                PRNTMCG      162     A
PRNTMCG            PRINT 226,U                                                  PRNTMCG      163     A
PRNTMCG   226      FORMAT(I5)                                                   PRNTMCG      164     A
PRNTMCG   230      CONTINUE                                                     PRNTMCG      165     A
PRNTMCG            PRINT 31                                                     PRNTMCG      166     A
PRNTMCG   240      LINECM = 0                                                   PRNTMCG      167     A
PRNTMCG                                                                         PRNTMCG      168     A
PRNTMCG   *        NOW RUN SOME CONSISENCY CHECKS                               PRNTMCG      169     A
PRNTMCG   *        REG( TXT(RVT(I))) = I - 1                                    PRNTMCG      170     A
PRNTMCG                                                                         PRNTMCG      171     A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PRNTMCG

```
PRNTMCG            RUSE = 0                                                    PRNTMCG    172    A
PRNTMCG            DO 250 I = 2,24                                             PRNTMCG    173    A
PRNTMCG            IF( RVT(I) .EQ. 0 ) GO TO 250                               PRNTMCG    174    A
PRNTMCG            REGNO = LINK( TXT( RVT(I) + 4 ) )                           PRNTMCG    175    A
PRNTMCG            IF( REGNO.EQ. I - 1 ) GO TO 250                             PRNTMCG    176    A
PRNTMCG            IF( REGNO .EQ. 10B ) GO TO 250                              PRNTMCG    177    A
PRNTMCG            K = RVT(I)                                                  PRNTMCG    178    A
PRNTMCG            PRINT 246,I-1,REGNO,TXT(K+4),K                              PRNTMCG    179    A
PRNTMCG   246      FORMAT("  ERROR, REG IN RVT("O2") IS "O2,2O22)              PRNTMCG    180    A
PRNTMCG            RUSE = 1                                                    PRNTMCG    181    A
PRNTMCG   250      CONTINUE                                                    PRNTMCG    182    A
PRNTMCG                                                                        PRNTMCG    183    A
PRNTMCG                                                                        PRNTMCG    184    A
PRNTMCG   *        MAKE SURE THAT *AXR* IS CORRECT                             PRNTMCG    185    A
PRNTMCG                                                                        PRNTMCG    186    A
PRNTMCG            DO 260 I = 2,24                                             PRNTMCG    187    A
PRNTMCG            IF( RBV(I) .EQ. 0 ) GO TO 260                               PRNTMCG    188    A
PRNTMCG            BIT = AXR .A. RBV(I)                                        PRNTMCG    189    A
PRNTMCG            IF( RVT(I) .NE. 0 ) GO TO 255                               PRNTMCG    190    A
PRNTMCG            IF( BIT .NE. 0 ) GO TO 260                                  PRNTMCG    191    A
PRNTMCG            PRINT 251,I-1                                               PRNTMCG    192    A
PRNTMCG   251      FORMAT(" ** ERROR **, RVT("O2") = 0 AND REG NOT AVAIL")     PRNTMCG    193    A
PRNTMCG            RUSE = 1                                                    PRNTMCG    194    A
PRNTMCG            GO TO 260                                                   PRNTMCG    195    A
PRNTMCG                                                                        PRNTMCG    196    A
PRNTMCG   255      IF( BIT .EQ. 0 ) GO TO 260                                  PRNTMCG    197    A
PRNTMCG            PRINT 256,I-1                                               PRNTMCG    198    A
PRNTMCG   256      FORMAT(" **ERROR **, RVT("O2*) " 0 AND REG IS AVAIL*)       PRNTMCG    199    A
PRNTMCG            RUSE = 1                                                    PRNTMCG    200    A
PRNTMCG   260      CONTINUE                                                    PRNTMCG    201    A
PRNTMCG            IF( RUSE .EQ. 0 ) RETURN                                    PRNTMCG    202    A
PRNTMCG            NRUSEE = NRUSEE + 1                                         PRNTMCG    203    A
PRNTMCG            IF( NRUSEE .LE. 5 ) CALL REMARK( 22L -REG USEAGE ERROR-      )PRNTMCG  204    A
PRNTMCG            RETURN                                                      PRNTMCG    205    A
PRNTMCG                                                                        PRNTMCG    206    A
PRNTMCG   **       PRINT FUNIT AND X-REG AVAIL TIMES                           PRNTMCG    207    A
PRNTMCG                                                                        PRNTMCG    208    A
PRNTMCG   300      DO 310 I = 1,9                                              PRNTMCG    209    A
PRNTMCG   310      FUNITS(2,I) = TFU(I+2)                                      PRNTMCG    210    A
PRNTMCG            DO 320 I = 1,8                                              PRNTMCG    211    A
PRNTMCG            XREGT(1,I) = REGS(I+16)                                     PRNTMCG    212    A
PRNTMCG   320      XREGT(2,I) = TRA(I+16)                                      PRNTMCG    213    A
PRNTMCG            PRINT 321,LAB,FUNITS,XREGT,LST                              PRNTMCG    214    A
PRNTMCG   321      FORMAT(A6, 9(A5,I5)/6X,8(2X,R2,I6)," LST "I5," LLT"I6)      PRNTMCG    215    A
PRNTMCG            RETURN                                                      PRNTMCG    216    A
PRNTMCG                                                                        PRNTMCG    217    A
PRNTMCG   **       FINAL CHECK                                                 PRNTMCG    218    A
PRNTMCG   *        ENSURE JLR,LXR,NSTORES = 0 AND RVT(I) = 0 FOR I = 0 TO 23   PRNTMCG    219    A
PRNTMCG   *        PRINT OUT SEQTIME AND MCG TIME                              PRNTMCG    220    A
PRNTMCG                                                                        PRNTMCG    221    A
PRNTMCG   400      LINECM = LINECM(2)                                          PRNTMCG    222    A
PRNTMCG            CALL SETST( 0 )                                             PRNTMCG    223    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN PRNTMCG

```
PRNTMCG            PRINT 401,MAXEST,DT                                          PRNTMCG      224     A
PRNTMCG   401      FORMAT(/* MCG SUMMARY, MAX(EST),DELTA T =*2I4,2X)            PRNTMCG      225     A
PRNTMCG                                                                         PRNTMCG      226     A
PRNTMCG            IF( NSTORES .GE. 0 ) GO TO 403                               PRNTMCG      227     A
PRNTMCG            PRINT 402,"N.ST = ", -NSTORES                                PRNTMCG      228     A
PRNTMCG   402      FORMAT(A10,O20)                                              PRNTMCG      229     A
PRNTMCG   403      IF( JLR .EQ. 0 ) GO TO 404                                   PRNTMCG      230     A
PRNTMCG            PRINT 402, "  JLR = ",JLR                                    PRNTMCG      231     A
PRNTMCG   404      IF( LXR .EQ. 0 ) GO TO 405                                   PRNTMCG      232     A
PRNTMCG            PRINT 402, "  LXR = ",LXR                                    PRNTMCG      233     A
PRNTMCG   405      IF( (AXR.A.377777B) .EQ. 377777B ) GO TO 410                 PRNTMCG      234     A
PRNTMCG            PRINT 402 , "  AXR = " , AXR                                 PRNTMCG      235     A
PRNTMCG                                                                         PRNTMCG      236     A
PRNTMCG   410      RVTE = 0                                                     PRNTMCG      237     A
PRNTMCG            DO 415 I = 1,24                                              PRNTMCG      238     A
PRNTMCG   415      RVTE = RVTE + RVT(I)                                         PRNTMCG      239     A
PRNTMCG            IF( RVTE .EQ. 0 ) GO TO 420                                  PRNTMCG      240     A
PRNTMCG            PRINT 411                                                    PRNTMCG      241     A
PRNTMCG   411      FORMAT(*  RVT " 0 *)                                         PRNTMCG      242     A
PRNTMCG                                                                         PRNTMCG      243     A
PRNTMCG  *                COMPUTE TIME FOR A SEQUENTIAL MACHINE                 PRNTMCG      244     A
PRNTMCG   420      T64 = -1                                                     PRNTMCG      245     A
PRNTMCG            DO 425 I = 2,NINST                                           PRNTMCG      246     A
PRNTMCG   425      T64 = T64 + FT( TXT(4*I-1) )                                 PRNTMCG      247     A
PRNTMCG                                                                         PRNTMCG      248     A
PRNTMCG   430      PRINT 426, SEQTIME,T64,T                                     PRNTMCG      249     A
PRNTMCG   426      FORMAT(/" *********"," PERT TIME, T(6400), T = ",3I6/)       PRNTMCG      250     A
PRNTMCG            LINECM = 0                                                   PRNTMCG      251     A
PRNTMCG            HEADER = 0                                                   PRNTMCG      252     A
PRNTMCG            RETURN                                                       PRNTMCG      253     A
PRNTMCG                                                                         PRNTMCG      254     A
PRNTMCG  **        HEAD OPTION - SET LISTING HEADER FLAG AND SUB TITLE          PRNTMCG      255     A
PRNTMCG                                                                         PRNTMCG      256     A
PRNTMCG   500      HEADER = 1                                                   PRNTMCG      257     A
PRNTMCG            CALL SETST( 6 , SUBTTL )                                     PRNTMCG      258     A
PRNTMCG            LIX = 0                                                      PRNTMCG      259     A
PRNTMCG            MAXEST = 0                                                   PRNTMCG      260     A
PRNTMCG            GO TO 101                                                    PRNTMCG      261     A
PRNTMCG                                                                         PRNTMCG      262     A
PRNTMCG            END                                                          PRNTMCG      263     A
PRNTMCG  *ENDIF                                                                 PRNTMCG      264     A
```

SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - PRNTMCG

IDENTIFIER	TOTAL	ACTIVE
PRNTMCG	264	264

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA      *DECK     CFA                                                          CFA            1     A
CFA                IDENT  CFA                                                   CFA            2     A
```

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA        CFA      TITLE  CFA - CONTROL FLOW ANALYSIS ROUTINES                    CFA          3     A
CFA       *CALL     SSTCALL                                                         CFA          4     A
CFA        B=CFA    RPVDEF                                                          CFA          5     A
CFA                 PASS2TM                                                         CFA          6           I
          -CCG1
CFA                                                                                 CFA          7           I
          -CCG1
CFA       *IF       DEF,CFA,1                                                       CFA          8     A
CFA                 TRACER GPO                                                      CFA          9     A
CFA        CFA      SPACE  2                                                        CFA         10     A
CFA       **        CFA - CONTROL FLOW ANALYSIS ROUTINES ( OPT=2 )                  CFA         11     A
CFA       *         CFA PROCESSES THE CONTROL FLOW INFORMATION FORMED IN *PRE* TO   CFA         12     A
CFA       *         FIND THE LOOPS IN THE PROGRAM, AND TO BUILD THE GRAPH/INTERVAL  CFA         13     A
CFA       *         STRUCTURE TABLES WHICH ARE USED BY THE OPTIMIZERS ( GPO/GRA )   CFA         14     A
CFA       *         TO DIRECT THE GLOBAL OPTIMIZATION.                              CFA         15     A
CFA       *                                                                         CFA         16     A
CFA       *         S.I. JASIK - SEPT 11, 1973                                      CFA         17     A
CFA                                                                                 CFA         18     A
CFA                 LIST   -R,X                                                     CFA         19     A
CFA                                                                                 CFA         20     A
CFA                 ENTRY  DGS,F.CFA                                                CFA         21           I
          -CCGNC
CFA                                                                                 CFA         22           I
          -CCGNC
CFA                 TABLES GST,CFT,BST,GLT,OTI,RND                                  CFA         23           I
          -CCG1     -CCGNC
CFA                 ECHO   1,Z=(BST,CFT,GST,GLT,OTI,RND)                            CCG1       354           I
          -CCGNC
CFA                 EXT    O.Z,L.Z                                                  CCG1       355           I
          -CCGNC
CFA                 TABLES (BST,CFT,GST,GLT,LCT,OTI,RND)                            CCGNC      809           I
          -CCG18
CFA                 TABLES (BST,CFT,GST,GLT,OTI,RND)  & LCT                         CCG18      178     A
CFA        O.BIT    EQU    O.BST                                                    CFA         24     A
CFA        L.BIT    EQU    L.BST                                                    CFA         25     A
CFA                 EXT    CBN,M.GRL,N.DG,NPB                                       CFA         26           I
          -CCG1
CFA                 EXT    M.GRL,N.DG,NPB                                           CCG1       356           I
          -CCGNC
CFA                 EXT#   (GIC,M.GRL,M.LP,N.DG,NPB,O.IST)                          CCGNC      810     A
CFA                                                                                 CFA         27     A
CFA        MAX.GST  EQU    1400B       MAX GRAPH SPACE ( INCORE )                  CFA         28     A
CFA                                                                                 CFA         29     A
CFA       **        DEBUG FACILITIES                                                CFA         30     A
CFA       *                                                                         CFA         31     A
CFA       *         TRACER (FGS,FGS.X,FIS1,FIS.X,CBD.X,FDG.X)                       CFA         32     A
CFA        SCRATCH  SPACE  3                                                        CFA         33     A
CFA       *         SCRATCH CELLS USED DURING GRAPH / INTERVAL FORMATION            CFA         34     A
CFA                                                                                 CFA         35     A
CFA                 USE    /TABLES/                                                 CFA         36           I
          -CCG1
CFA                 USE    /CCG.SCR/                                                CCG1       357           I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

                -CCG18
        CFA                USE    /CCGSCR/                                          CCG18        179     A
        CFA       PBN      BSS    1           PREVIOUS BLOCK NUMBER ( *CBN* ON ENTRY )  CFA           37     A
        CFA                                                                          CFA           38          I
                -CCG7
        CFA       LB       BSS    2           LINK BACKWARDS / L.GST AT START OF GRAPH  CFA           39     A
        CFA       N.BLK    BSS    1           NUMBER OF BLOCKS IN GRAPH                 CFA           40     A
        CFA                                                                          CFA           41          I
                -CCG7
        CFA       LC       BSS    1           SAVED LENGTH OF *CFT*                     CFA           42     A
        CFA       EIB      BSS    2           EDGE INDEX BASE                           CFA           43     A
        CFA       EII      EQU    EIB         EDGE INDEX INDEX  ( RELATIVE TO O.GST )   CFA           44     A
        CFA       ETB      EQU    EIB+1       EDGE TABLE BASE                           CFA           45     A
        CFA       ILB      BSS    1           INTERVAL LIST BASE                        CFA           46     A
        CFA                                                                          CFA           47     A
        CFA       HTL      BSS    1           HEADER TABLE LIST                         CFA           48     A
        CFA       NPH      BSS    1           N.PRED OF HEADER AT BEGIN OF INTERVAL     CFA           49     A
        CFA       FTH      BSS    1           FALL THROUGH TO HEADER FROM A LATCHING NODE  FCC0408     92     A
        CFA       SCR      BSS    1           "0 IF GRAPH CONTAINS LOOPS                CFA           50     A
        CFA       LINT     BSS    1                                                     CFA           51          I
                -FCC0392
        CFA       IP       BSS    1           INTERVAL POINTER ( CBD )                  CFA           52     A
        CFA       NN       BSS    1           N.NODES REMAINING ( CBD12 )               CFA           53     A
        CFA       I        EQU    NN                                                    CFA           54     A
        CFA                USE    0                                                     CFA           55     A
        CFA                SPACE  2                                                     CFA           56     A
        CFA      *CALL     CFTBLD                                                       CFA           57     A
        CFA       TMP      SPACE  3                                                     CFA           58     A
        CFA      **        I. - TEMPORARY TABLE FOR INTERVAL FORMATION                  CFA           59     A
        CFA                                                                          CFA           60     A
        CFA                DESCRIBE I.,60                                               CFA           61     A
        CFA       INI      DEFINE 1            IN AN INTERVAL                           CFA           62     A
        CFA       INH      DEFINE 1            IN HEADER LIST                           CFA           63     A
        CFA                DEFINE 4                                                     CFA           64          I
                -FCC1191
        CFA       1P       DEFINE 1            =1 IF ONLY 1 PREDECESSOR                 FCC1191        6     A
        CFA                DEFINE 3                                                     FCC1191        7     A
        CFA       HN       DEFINE 18           HEADER NODE NUMBER                       CFA           65     A
        CFA       IA       DEFINE 18           ADDRESS OF NODE IN INTERVAL LIST         CFA           66     A
        CFA       PC       DEFINE 18           PREDECESSOR COUNT                        CFA           67     A
        CFA                SPACE  2                                                     CFA           68     A
        CFA      **        I. - TEMP TABLE FOR BACK DOMINATOR CALCULATION               CFA           69     A
        CFA      *         NOTE - IF NODES IN INTERVAL @ 60 , THEN BITN = NI            CFA           70     A
        CFA                                                                          CFA           71     A
        CFA                DESCRIBE I.,60                                               CFA           72     A
        CFA       BITN     DEFINE 12           PACKED BIT NUMBER                        CFA           73     A
        CFA       WI       DEFINE 18           WORD INDEX                               CFA           74     A
        CFA       BVI      DEFINE 18           BIT VECTOR INDEX                         CFA           75     A
        CFA       NI       DEFINE 12           NODE INDEX ( REL TO INTERVAL BASE )      CFA           76     A
        CFA                SPACE  2                                                     CFA           77     A
        CFA                LIST   R                                                     CFA           78     A
        CFA                QUAL   CODE                                                  CFA           79     A

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA       CGT      TITLE  CGT - CHECK GRAPH TABLE LENGTH AND FLUSH IF NECESSARY          CFA        80    A
CFA      **        CGT - CHECK GRAPH TABLE LENGTH AND FLUSH TO DISK IF TOO BIG           CFA        81    A
CFA                                                                                      CFA        82    A
CFA       CGT      ROUTINE                                                               CFA        83    A
CFA                SA1    L.GST                                                          CFA        84    A
CFA                SX2    X1-MAX.GST                                                     CFA        85    A
CFA                MI     X2,CGT      IF L.GST < MAX.GRS  */ SMALL ENOUGH FOR CORE       CFA        86    A
CFA                SA5    N.DG                                                           CFA        87    A
CFA                ZR     X5,CGT      IF N.DG = 0   */ NO DERIVED GRAPHS                 CFA        88    A
CFA                                                                                      CFA        89    A
CFA      *         FLUSH GRAPHS TO DISK, SETUP RANDOM INDEX IN WORKING STORAGE.          CFA        90    A
CFA                                                                                      CFA        91    A
CFA                SA3    O.GST                                                          CFA        92    A
CFA                SX4    X3+B1       FWA = O.GST + 1                                    CFA        93    A
CFA                SA5    X4                                                             CFA        94    A
CFA                LX5    -GA.LFP     LEN = LF[FWA]                                      CFA        95    A
CFA                MX6    0                                                              CFA        96    A
CFA                SX7    B1                                                             CFA        97    A
CFA                SA6    I           I = 0                                              CFA        98    A
CFA                SA7    =XGIC       GIC = 1       */ INDEX                             CFA        99    A
CFA                                                                                      CFA       100    A
CFA       CGT1     SA2    GIC                                                            CFA       101    A
CFA                IX7    X2+X5       GIC = GIC + LEN                                    CFA       102    A
CFA                SA7    A2                                                             CFA       103    A
CFA                CALL   SBD         SAVE GRAPH ON DISK                                 CFA       104          I
          -CCGNC
CFA                CALL   SBD#        SAVE GRAPH ON DISK                                 CCGNC     811    A
CFA                SA5    =XO.IST                                                        CFA       105    A
CFA                BX7    X1                                                             CFA       106    A
CFA                SB2    X5                                                             CFA       107    A
CFA                SA1    I                                                              CFA       108    A
CFA                SA7    B2+X1       IST(I) = RIW  */ SAVE MASS STORAGE ADDRESS         CFA       109    A
CFA                SX6    X1+B1       I = I + 1                                          CFA       110    A
CFA                SA3    O.GST                                                          CFA       111    A
CFA                SA5    GIC                                                            CFA       112    A
CFA                IX4    X3+X5       FWA = O.GST + GIC                                  CFA       113    A
CFA                SA2    X4                                                             CFA       114    A
CFA                LX2    -GA.LFP                                                        CFA       115    A
CFA                SX5    X2          LEN = LF[FWA]                                      CFA       116    A
CFA                SA6    A1                                                             CFA       117    A
CFA                NZ     X5,CGT1     IF LEN " 0                                         CFA       118    A
CFA                                                                                      CFA       119    A
CFA                SA4    =XM.GRL                                                        CFA       120    A
CFA                SX6    B2                                                             CFA       121    A
CFA                SX7    A7+B1                                                          CFA       122    A
CFA                SA6    GIC         GIC = O.IST   */ POINT TO FIRST GRAPH              CFA       123    A
CFA                SA7    O.IST       O.IST = O.IST + I                                  CFA       124    A
CFA                SX6    X4+1                                                           CFA       125    A
CFA                SA6    L.GST       L.GST = M.GRL + 1                                  CFA       126    A
CFA                EQ     CGT                                                            CFA       127    A
CFA       DGS      TITLE  DGS - DERIVE GRAPH STRUCTURE                                   CFA       128    A
CFA      **        DGS - DERIVE GRAPH STRUCTURE ( CONTROL FLOW INFORMATION )             CFA       129    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA                                                                                  CFA        130    A
CFA      DGS0     ADDWRD GST,B0      ADDWRD( GST , 0 )                               CFA        131    A
CFA               SA6    L.CFT       L.CFT = 0                                       CFA        132    A
CFA                                                                                  CFA        133    A
CFA               RJ     CGT         CHECK GRAPH TABLES                              CFA        134    A
CFA                                                                                  CFA        135    A
CFA      DGS      ROUTINE                                                             CFA        136    A
CFA               SX6    1                                                           CCG27       70    A
CFA               SA6    =XCC$OPF    CC$OPF = 1    */ MARK END OF BRIDGE PHASE       CCG27       71    A
CFA               CALL   AFT         ADJUST FLOW TABLE                               CFA        137    A
CFA               MX6    0                                                           CFA        138    A
CFA               SA6    L.GST       L.GST = 0                                       CFA        139    A
CFA               ADDWRD GST,B0      ADDWRD( GST , 0 )                               CFA        140    A
CFA               SA5    CBN                                                         CFA        141         I
         -CCG1
CFA               SA5    =XCG.CBN                                                    CCG1       358         I
         -CCGNC
CFA               SA5    =XCC$CBN                                                    CCGNC      812    A
CFA               MX6    0                                                           CFA        142    A
CFA               BX7    X5                                                          CFA        143    A
CFA               SA6    LB          LB = 0        */ LINK BACKWARDS                 CFA        144    A
CFA               SA7    PBN         PBN = CBN       */ CBN AT BEGIN OF GRAPH        CFA        145         I
         -CCG1
CFA               SA7    PBN         PBN = CG.CBN    */ CBN AT BEGIN OF GRAPH        CCG1       359    A
CFA               SX7    X7-1                                                        CFA        146    A
CFA               SA7    N.BLK       N.BLK = CBN - 1 */ NUMBER OF BLOCKS             CFA        147    A
CFA               SA7    NPB         NPB  = CBN - 1  */ N.PROGRAM BLOCKS             CFA        148    A
CFA                                                                                  CFA        149    A
CFA               RJ     FGS         FORM GRAPH STRUCTURE                            CFA        150    A
CFA               RJ     FIS         FORM INTERVAL STRUCTURE                         CFA        151    A
CFA               RJ     MRB         MARK REACHABLE BLOCKS                           CFA        152    A
CFA                                                                                  CFA        153    A
CFA      *        FORM DERVIED GRAPH OF CURRENT GRAPH IT IT HAS LOOPS                CFA        154    A
CFA                                                                                  CFA        155    A
CFA       DGS1    SA1    SCR                                                         CFA        156    A
CFA               ZR     X1,DGS0     IF SCR = 0    */ NO LOOPS                       CFA        157    A
CFA                                                                                  CFA        158    A
CFA               RJ     FDG         FORM DERIVED GRAPH                              CFA        159    A
CFA               SA1    N.DG                                                        CFA        160    A
CFA               SA2    CBN                                                         CFA        161         I
         -CCG1
CFA               SA2    CG.CBN                                                      CCG1       360         I
         -CCGNC
CFA               SA2    CC$CBN                                                      CCGNC      813    A
CFA               SX6    X1+B1       N.DG = N.DG + 1                                 CFA        162    A
CFA               LX7    X2                                                          CFA        163    A
CFA               SA7    PBN         PBN = CBN                                       CFA        164    A
CFA               SA6    A1                                                          CFA        165    A
CFA                                                                                  CFA        166    A
CFA               RJ     FGS         FORM GRAPH STRUCTURE TABLES                     CFA        167    A
CFA               RJ     FIS         FORM INTERVAL STRUCTURE                         CFA        168    A
CFA               EQ     DGS1                                                        CFA        169    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA      AFT      TITLE  AFT - ADJUST FLOW TABLE                                    CFA      170    A
CFA     **        AFT - ADJUST FLOW TABLE TO REPLACE *IH* ORDINALS WITH BLOCK       CFA      171    A
CFA     *                NUMBERS IN *TO* FIELD ( FORWARD REFERENCES ).             CFA      172    A
CFA                                                                                CFA      173    A
CFA      AFT      ROUTINE                                                           CFA      174         I
        -CCG27
CFA               SA1    L.CFT                                                      CFA      175         I
        -CCG27
CFA               SA2    O.CFT                                                      CFA      176         I
        -CCG27
CFA               SA3    O.SYM                                                      CFA      177         I
        -CCG1     -CCG27
CFA               SA3    =XO.STM                                                    CCG1     361         I
        -CCG7     -CCG27
CFA               SA3    =XO.SYM                                                    CCG7     134         I
        -CCG27
CFA               SA4    O.GLT                                                      CFA      178         I
        -CCG27
CFA               SA0    X3-1                                                       CFA      179         I
        -CCG1     -CCG27
CFA               SA0    X3+                                                        CCG1     362         I
        -CCG27
CFA               SB2    X4-I.GL-1                                                  CFA      180         I
        -CCG18    -CCG27
CFA               SB2    X4-I.GL                                                    CCG18    180         I
        -CCG27
CFA               SB3    X2          I = O.CFT                                      CFA      181         I
        -CCG27
CFA               MX0    -CF.FROMP                                                  CFA      182         I
        -CCG27
CFA               SB4    B3+X1       L = I + L.CFT                                  CFA      183         I
        -CCG27
CFA               MX7    -12                                                        CFA      184         I
        -CCG27
CFA               SB6    I.GL                                                       CFA      185         I
        -CCG27
CFA               EQ     AFT2                                                       CFA      186         I
        -CCG27
CFA                                                                                CFA      187         I
        -CCG27
CFA      AFT1     SA4    B2+B5       BN = GLT(H-1)                                  CFA      188         I
        -CCG27
CFA               IX6    X6+X4                                                      CFA      189         I
        -CCG27
CFA               SA6    A5+         TO[CI] = BN                                    CFA      190         I
        -CCG27
CFA                                                                                CFA      191         I
        -CCG27
CFA      AFT2     SA5    B3          CI = I                                         CFA      192         I
        -CCG27
CFA               GE     B3,B4,AFT   IF I \ L      */ END OF TABLE                  CFA      193         I
        -CCG27

        0         1         2         3         4         5         6         7         8
        12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA              LX5    59-CF.JPP                                                  CFA          194          I
         -CCG27
CFA              SB3    B3+B1       I = I + 1                                      CFA          195          I
         -CCG27
CFA              PL     X5,AFT2     IF ^JP[CI]    */ *TO* IS A BLOCK NUMBER        CFA          196          I
         -CCG27
CFA              LX5    1+CF.JPP                                                   CFA          197          I
         -CCG27
CFA              SB5    X5          H = TO[CI]                                     CFA          198          I
         -CCG27
CFA       AFT0   SX6    B4+                                                        CCG27         72     A
CFA              SA6    L.CFT       L.CFT = J                                      CCG27         73     A
CFA                                                                               CCG27         74     A
CFA       AFT    ROUTINE                                                           CCG27         75     A
CFA              ADDWRD CFT,-B1     ADDWRD( CFT , -1 )  */ LOOP TERMINATOR         CCG27         76     A
CFA              SB3    X2                (B3) = CFT                                CCG27         77     A
CFA              SB4    B0          J = 0         */ STORE INDEX                   CCG27         78     A
CFA              SA3    =XO.SYM                                                   CCG27         79     A
CFA              SA4    O.GLT                                                     CCG27         80     A
CFA              MX0    -CF.FROMP                                                 CCG27         81     A
CFA              SA0    X3                                                        CCG27         82     A
CFA              SB2    X4-I.GL                                                   CCG27         83     A
CFA              SB6    I.GL                                                      CCG27         84     A
CFA              MX7    -12                                                       CCG27         85     A
CFA              SA5    B3-B1       CI = CFT - 1                                  CCG27         86     A
CFA              EQ     AFT3                                                      CCG27         87     A
CFA                                                                               CCG27         88     A
CFA       AFT1   SA4    B2+B5       BN = GLT(H-1)                                 CCG27         89     A
CFA              IX5    X6+X4       TO[CIW] = BN                                  CCG27         90     A
CFA                                                                               CCG27         91     A
CFA       AFT2   BX6    X5                                                        CCG27         92     A
CFA              SA6    B3+B4       CFT(J) = CIW                                  CCG27         93     A
CFA              SB4    B4+1        J = J + 1                                     CCG27         94     A
CFA                                                                               CCG27         95     A
CFA       AFT3   SA5    A5+B1       CI = CI + 1;  CIW = [CI]                      CCG27         96     A
CFA              LX5    59-CF.JPP                                                 CCG27         97     A
CFA              ZR     X5,AFT3     IF CIW = 0    */ DELETED ENTRY                CCG27         98     A
CFA              PL     X5,AFT2     IF ^JP[CI]    */ *TO* IS A BLK NUMBER         CCG27         99     A
CFA              LX5    1+CF.JPP                                                  CCG27        100     A
CFA              SB5    X5          H = TO[CI]                                    CCG27        101     A
CFA              MI     B5,AFT0     IF TO[CI] < 0 */ END OF THE TABLE             CCG27        102     A
CFA              BX6    X0*X5                                                     CFA          199     A
CFA              GE     B5,B6,AFT1  IF H \ I.GL                                   CFA          200     A
CFA                                                                               CFA          201     A
CFA              SB5    B5+B5                                                     CFA          202          I
         -CCG1
CFA              SA4    A0-B5       WORDB = [O.SYM-1-2*H]                         CFA          203          I
         -CCG1
CFA              SB7    A0+B5                                                     CCG1         363     A
CFA              SB5    B5+B5                                                     CCG1         364     A
CFA              SA4    B7+B5       WORDA = SYM(3*IH)                             CCG1         365     A
CFA              BX3    -X7*X4                                                    CFA          204     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA                IX6    X6+X3       TO[CI] = BN[WORDB]                          CFA          205          I
          -CCG1
CFA                IX6    X6+X3       TO[CI] = BN[WORDA]                          CCG1         366          I
          -CCG27
CFA                SA6    A5+                                                     CFA          206          I
          -CCG27
CFA                IX5    X6+X3       TO[CIW] = BN[WORDA]                         CCG27        103     A
CFA                EQ     AFT2                                                    CFA          207     A
CFA        LABD    SPACE  3,14                                                    CCG16        263     A
CFA       **       CG$LABD - DEFINE BLOCK NUMBER ASSOCIATED WITH A LABEL          CCG16        264     A
CFA       *                                                                       CCG16        265     A
CFA       *        ENTRY  (X1) = IH, SYMTAB ORDINAL OF LABEL DEFINITION           CCG16        266     A
CFA       *                (CC$CBN) = CURRENT BLOCK NUMBER                        CCG16        267     A
CFA                                                                               CCG16        268     A
CFA        LABD1   SA2    =XO$GLT                                                 CCG16        269     A
CFA                BX6    X5                                                      CCG16        270     A
CFA                SA6    X2+B2       GLT(H[IH]) = CBN                            CCG16        271     A
CFA                                                                               CCG16        272     A
CFA        CG$LABD ROUTINE                                                        CCG16        273     A
CFA                SA5    CC$CBN                                                  CCG16        274     A
CFA                SB2    X1-I.GL                                                 CCG16        275     A
CFA                PL     B2,LABD1    IF I[IH] " 0  */ A GL                       CCG16        276     A
CFA                SA2    O.SYM                                                   CCG16        277     A
CFA                SB2    X1                                                      CCG16        278     A
CFA                LX1    1                                                       CCG16        279     A
CFA                IX3    X1+X2                                                   CCG16        280     A
CFA                SA4    X3+B2                                                   CCG16        281     A
CFA                MX0    -WA.BNL                                                 CCG16        282     A
CFA                BX6    X0*X4                                                   CCG16        283     A
CFA                BX7    X6+X5                                                   CCG16        284     A
CFA                SA7    A4          BN[WA] = CBN                                CCG16        285     A
CFA                EQ     CG$LABD                                                 CCG16        286     A
CFA        MRB     TITLE  MRB - MARK REACHABLE BLOCKS                             CFA          208     A
CFA       **       MRB - MARK REACHABLE BLOCKS                                    CFA          209     A
CFA       *        SET *RB* BIT IN *BIT* FOR BLOCKS THAT ARE IN INTERVALS.        CFA          210     A
CFA                                                                               CFA          211     A
CFA        MRB     ROUTINE                                                        CFA          212     A
CFA                SA1    O.GST                                                   CFA          213     A
CFA                SA2    ILB                                                     CFA          214     A
CFA                MX0    1                                                       CFA          215     A
CFA                LX0    1+BI.RBP                                                CFA          216     A
CFA                IX7    X1+X2                                                   CFA          217     A
CFA                SA3    N.BLK                                                   CFA          218     A
CFA                SA4    O.BIT                                                   CFA          219     A
CFA                SB2    X3          NB = N.BLK                                  CFA          220     A
CFA                SB3    X4                                                      CFA          221     A
CFA                SA5    X7          IP = O.GST + ILB                            CFA          222     A
CFA                                                                               CFA          223     A
CFA        MRB1    LX5    -IP.NODEP                                               CFA          224     A
CFA                SB4    X5          N = NODE[IP]                                CFA          225     A
CFA                SB2    B2-B4       NB = NB - N                                 CFA          226     A
CFA                SA5    A5+B1       IP = IP + 1                                 CFA          227     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA                                                                                  CFA        228    A
CFA      MRB2     LX5    -IM.BNP                                                     CFA        229    A
CFA               SX7    X5                                                          CFA        230    A
CFA               LX7    1                                                           CFA        231    A
CFA               SA4    B3+X7                                                       CFA        232    A
CFA               BX6    X0+X4        RB[BIT(BN[IP])] = 1                            CFA        233    A
CFA               SA5    A5+B1        IP = IP + 1                                    CFA        234    A
CFA               SB4    B4-1         N = N - 1                                      CFA        235    A
CFA               SA6    A4                                                          CFA        236    A
CFA               NZ     B4,MRB2      IF N " 0                                       CFA        237    A
CFA               NZ     X5,MRB1      IF [IP] " 0                                    CFA        238    A
CFA                                                                                 CFA        239    A
CFA      *        PROGRAM GRAPH CONTAINS UNREACHABLE BLOCKS.                         CFA        240    A
CFA      *        REFORM THE EDGE TABLES AND THE INTERVALS, SINCE THE DEAD CODE MAY  CFA        241    A
CFA      *        MAY CAUSE THE FORMATION OF SOME SUB-MAXIMAL INTERVALS.             CFA        242    A
CFA                                                                                 CFA        243    A
CFA               ZR     B2,MRB       IF NB = 0     */ ALL BLOCKS REACHABLE          CFA        244    A
CFA               SA1    LC                                                          CFA        245    A
CFA               MX6    0                                                           CFA        246    A
CFA               SA6    L.CFT        L.CFT = 0                                      CFA        247    A
CFA               SA6    =XL.LCT      L.LCT = 0                                      FCC1319    136         I
         -CCG18
CFA               SA6    =XL.LCT#     L.LCT = 0                                      CCG18      181    A
CFA               SA6    LB           LB = 0                                         CFA        248    A
CFA               SA6    =XN.FTH#     N.FTH = 0                                      CCGA021     62    A
CFA               ALLOC  CFT,X1       ALLOC( CFT , LC )                              CFA        249    A
CFA               SA1    O.BIT                                                       CFA        250    A
CFA               SB2    X2           C0 = O.CFT                                     CFA        251    A
CFA               SA5    B2-B1                                                       CFA        252    A
CFA               BX7    X5                                                          CFA        253    A
CFA               SA7    A5           CI = C0 - 1                                    CFA        254    A
CFA               SB4    B1           BN = 1                                         CFA        255    A
CFA               SA4    N.BLK                                                       CFA        256    A
CFA               SA3    O.GST                                                       CFA        257    A
CFA               SB5    X4                                                          CFA        258    A
CFA               SB7    X3+3         EIB = O.GST + 3                                CFA        259    A
CFA               MX0    1+ET.SUCCL                                                  CFA        260    A
CFA               LX0    ET.SUCCL                                                    CFA        261    A
CFA               SA2    X1+2         BI = O.BIT + 2                                 CFA        262    A
CFA               SX1    B1                                                          CFA        263    A
CFA               LX1    CF.FROMP                                                    CFA        264    A
CFA               SX6    B1                                                          CFA        265    A
CFA               SA6    L.GST        L.GST = 1                                      CFA        266    A
CFA               SA6    =XHO$DC      HO$DC = 1     */ INDICATE DEAD CODE EXISTS     CCGNC      814    A
CFA               EQ     MRB4                                                        CFA        267    A
CFA                                                                                 CFA        268    A
CFA      *        UNREACHABLE BLOCK, SHOW FLOW TO EXIT NODE ( FOR *FGS* )            CFA        269    A
CFA                                                                                 CFA        270    A
CFA      MRB3     BX7    X1                                                          CFA        271    A
CFA               SA7    A7+B1        CI = CI + 1;  [CI] = CFTE( BN , 0 )            CFA        272    A
CFA               EQ     MRB6                                                        CFA        273    A
CFA                                                                                 CFA        274    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA       MRB4     LX2     59-BI.RBP                                                CFA          275     A
CFA                PL      X2,MRB3      IF ^RB[BI]                                  CFA          276     A
CFA                SA5     B7+B4        EI = EIB + BN                               CFA          277     A
CFA                LX5     -EI.SUIP                                                 CFA          278     A
CFA                UX6     B6,X5        NS = NS[EI]                                 CFA          279     A
CFA                SA4     B7+X5        SI = EIB + SUI[EI]                          CFA          280     A
CFA                                                                                 CFA          281     A
CFA       MRB5     BX3     X0*X4                                                    CFA          282     A
CFA                IX7     X1+X3                                                    CFA          283     A
CFA                SA7     A7+B1        CI = CI + 1;  [CI] = CFTE( BN , SUCC[SI] )  CFA          284     A
CFA                SB6     B6-B1        NS = NS - 1                                 CFA          285     A
CFA                SA4     A4+B1        SI = SI + 1                                 CFA          286     A
CFA                NZ      B6,MRB5      IF NS " 0                                   CFA          287     A
CFA                                                                                 CFA          288     A
CFA       MRB6     SB4     B4+B1        BN = BN + 1                                 CFA          289     A
CFA                SA2     A2+2         BI = BI + 2                                 CFA          290     A
CFA                SX1     B4                                                       CFA          291     A
CFA                LX1     CF.FROMP                                                 CFA          292     A
CFA                LE      B4,B5,MRB4   IF BN @ N.BLK                               CFA          293     A
CFA                                                                                 CFA          294     A
CFA                SB6     A7+B1                                                    CFA          295     A
CFA                SX6     B6-B2        L.CFT = CI + 1 - C0                         CFA          296     A
CFA                SA6     L.CFT                                                    CFA          297     A
CFA                SX7     B5+B1        CBN = N.BLK + 1                             CFA          298     A
CFA                SA7     CBN                                                      CFA          299         I
          -CCG1
CFA                SA7     CG.CBN                                                   CCG1         367         I
          -CCGNC
CFA                SA7     CC$CBN                                                   CCGNC        815     A
CFA                RJ      FGS          REFORM GRAPH STRUCTURE TABLES               CFA          300     A
CFA                RJ      FIS          REFORM INTERVALS                            CFA          301     A
CFA                                                                                 CFA          302     A
CFA       *        SCAN *BIT* AND LIST LINE NUMBERS OF UNREACHABLE BLOCKS           CFA          303         I
          -CCG26
CFA                                                                                 CFA          304         I
          -CCGNC   -CCG26
CFA                SA1     =XPROGNAM                                                CFA          305         I
          -CCG1    -CCGNC    -CCG26
CFA                SA1     =XCO.PRGN                                                CCG1         368         I
          -CCGNC   -CCG26
CFA                BX6     X1                                                       CFA          306         I
          -CCGNC   -CCG26
CFA                SA6     MRBD+2                                                   CFA          307         I
          -CCGNC   -CCG26
CFA                MESSAGE MRBD,,R    *   DEAD CODE IN - PROGNAM *                  CFA          308         I
          -CCG1    -CCGNC    -CCG26
CFA                MESSAGE MRBD,,R    *   DEAD CODE IN - CO.PRGN *                  CCG1         369         I
          -CCGNC   -CCG26
CFA                                                                                 CFA          309         I
          -CCGNC   -CCG26
CFA                SA1     =XCO.LO=S                                                FCC1623X       1         I
          -CCGNC   -CCG26

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA                 NZ     X1,MRB6A    IF LO=S " 0                                  FCC1623X       2          I
          -CCGNC    -CCG26
CFA                 SA1    =XR=FLAG                                                 CFA          310          I
          -CCG1     -CCGNC    -CCG26
CFA                 SA1    =XCO.LO=M                                                CCG1         370          I
          -CCGNC    -CCG26
CFA                 ZR     X1,MRB      IF NO MAP                                    CFA          311          I
          -CCGNC    -CCG26
CFA                                                                                 CFA          312          I
          -CCGNC    -CCG26
CFA                 LISTL  MRBB,1                                                   CFA          313          I
          -FCC1623X -CCGNC    -CCG26
CFA        MRB6A    LISTL  MRBB,1                                                   FCC1623X       3          I
          -CCGNC    -CCG26
CFA                 LISTL  MRBA,MRBB+1-MRBA                                         CFA          314          I
          -CCGNC    -CCG26
CFA                                                                                 CCGNC        816          I
          -CCG26
CFA                 SA1    =XHO$IE                                                  CCGNC        817          I
          -CCG26
CFA                 ZR     X1,MRB      IF HO$IE = 0  */ NO INFORMATIVE ERRORS       CCGNC        818          I
          -CCG26
CFA                                                                                 CCGNC        819          I
          -CCG26
CFA                 LISTL  MRBA,6,1    SPACE 1 AND LIST HEADER LINES                CCGNC        820          I
          -CCG26
CFA                 LISTL  MRBB,6                                                   CCGNC        821          I
          -CCG26
CFA                 SB6    4           I = 4                                        CFA          315          I
          -CCG26
CFA                                                                                 CFA          316          I
          -CCG26
CFA        MRB7     SA1    =1H                                                      CFA          317          I
          -CCG26
CFA                 SA3    L.BIT                                                    CFA          318          I
          -CCG26
CFA                 SA5    O.BIT                                                    CFA          319          I
          -CCG26
CFA                 BX6    X1                                                       CFA          320          I
          -CCG26
CFA                 SB7    X3                                                       CFA          321          I
          -CCG26
CFA                 SA6    MRBA                                                     CFA          322          I
          -CCG26
CFA                 SX0    -MRBC                                                    CFA          323          I
          -CCGNC    -CCG26
CFA                 SX0    -5                                                       CCGNC        822          I
          -CCG26
CFA                 EQ     B6,B7,MRB   IF I = L.BIT                                 CFA          324          I
          -CCG26
CFA                                                                                 CFA          325          I
          -CCG26

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA       MRB8     SA1    X5+B6       BIW = BIT(I)                                  CFA      326      I
          -CCG26
CFA                LX1    59-BI.RBP                                                 CFA      327      I
          -CCG26
CFA                PL     X1,MRB11    IF ^RB[BIW]                                   CFA      328      I
          -CCG26
CFA       MRB9     SB6    B6+2        I = I + 2                                     CFA      329      I
          -CCG26
CFA                LT     B6,B7,MRB8  IF I < L.BIT                                  CFA      330      I
          -CCG26
CFA                                                                                 CFA      331      I
          -CCG26
CFA       MRB10    RJ     LAL         LIST LAST LINE                                CFA      332      I
          -CCG26
CFA                EQ     MRB                                                       CFA      333      I
          -CCG26
CFA                                                                                 CFA      334      I
          -CCG26
CFA       MRB11    LX1    1+BI.RBP-BI.FLNP                                          CFA      335      I
          -CCG26
CFA                SX1    X1                                                        CFA      336      I
          -CCG26
CFA                CALL   CDD         CONVERT LINE NUMBER                           CFA      337      I
          -CCG1    -CCG26
CFA                CALL   CDD=         CONVERT LINE NUMBER                          CCG1     371      I
          -CCG26
CFA                SA6    A6+B1       STORE IN LINE BUFFER                          CFA      338      I
          -CCG26
CFA                SX0    X0+B1                                                     CFA      339      I
          -CCG26
CFA                MI     X0,MRB9     IF NOT A FULL LINE                            CFA      340      I
          -CCG26
CFA                                                                                 CFA      341      I
          -CCG26
CFA                RJ     LAL         LIST LINE                                     CFA      342      I
          -CCG26
CFA                EQ     MRB7                                                      CFA      343      I
          -CCG26
CFA                                                                                 CFA      344      I
          -CCG26
CFA       MRBA     DATA   H* STATEMENTS BEGINNING AT BELOW LINE NUMBERS ARE UNREA   CFA      345      I
          -CCGNC   -CCG26
CFA       ,CHABLE ( DEAD CODE ), AND WILL NOT BE PROCESSED*                         CFA      346      I
          -CCGNC   -CCG26
CFA       MRBB     DATA   8L                                                        CFA      347      I
          -CCGNC   -CCG26
CFA       MRBC     EQU    MRBB-MRBA-1                                               CFA      348      I
          -CCGNC   -CCG26
CFA       MRBD     DATA   28L     DEAD CODE IN - NAMEXXX                            CFA      349      I
          -CCGNC   -CCG26
CFA       MRBA     DATA   50H   STATEMENTS BEGINNING AT BELOW LINE NUMBERS ARE      CCGNC    823      I
          -CCG26

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA                 DATA   0                                                    CCGNC        824          I
            -CCG26
CFA          MRBB   DATA   51L  UNREACHABLE (DEAD CODE) AND WILL NOT BE PROCESSED  CCGNC        825          I
            -CCG26
CFA          LAL    SPACE  3           LAL                                       CFA          350          I
            -CCG26
CFA         **      LAL - LIST A LINE                                            CFA          351          I
            -CCG26
CFA                                                                              CFA          352          I
            -CCG26
CFA          LAL    ROUTINE                                                      CFA          353          I
            -CCG26
CFA                 SA4    MRBB                                                  CFA          354          I
            -CCGNC  -CCG26
CFA                 SX7    B6                                                    CFA          355          I
            -CCGNC  -CCG26
CFA                 BX6    X4                                                    CFA          356          I
            -CCGNC  -CCG26
CFA                 SX7    B6                                                    CCGNC        826          I
            -CCG26
CFA                 MX6    0                                                     CCGNC        827          I
            -CCG26
CFA                 SA6    A6+B1                                                 CFA          357          I
            -CCG26
CFA                 SB7    A6+B1                                                 CFA          358          I
            -CCG26
CFA                 SA7    I                                                     CFA          359          I
            -CCG26
CFA                 LISTL  MRBA,B7-MRBA                                          CFA          360          I
            -CCG26
CFA                 SA3    I                                                     CFA          361          I
            -CCG26
CFA                 SB6    X3+2        I = I + 2                                 CFA          362          I
            -CCG26
CFA                 EQ     LAL                                                   CFA          363          I
            -CCG26
CFA         *       SCAN *BIT* AND BUILD A TABLE OF UNREACHABLE BLOCKS IN *RND*  CCG26         51     A
CFA         *       FOR THE HOST TO PRINT OUT.                                   CCG26         52     A
CFA                                                                              CCG26         53     A
CFA                 SA1    =XHO$IE                                               CCG26         54     A
CFA                 ZR     X1,MRB      IF HO$IE = 0  */ NO INFORMATIVE ERRORS    CCG26         55     A
CFA                                                                              CCG26         56     A
CFA                 SA5    L.BIT                                                 CCG26         57     A
CFA                 AX1    B1,X5                                                 CCG26         58     A
CFA                 ALLOC  RND,X1      ALLOC( RND , L.BIT/2 )                    CCG26         59     A
CFA                 SA4    O.BIT                                                 CCG26         60     A
CFA                 SB3    B0          I = 0                                     CCG26         61     A
CFA                 SB4    4           J = 4                                     CCG26         62     A
CFA                 SB5    X5                                                    CCG26         63     A
CFA                                                                              CCG26         64     A
CFA          MRB8   SA3    X4+B4                                                 CCG26         65     A
CFA                 LX3    59-BI.RBP                                             CCG26         66     A

            0         1         2         3         4         5         6         7         8
            12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA              MI     X3,MRB9     IF RB[BIT(J)]   */ BLOCK IS REACHABLE          CCG26      67     A
CFA              SX6    X1                                                          CCG26      68          I
         -CCG27
CFA              LX3    1+BI.RBP-BI.FLNP                                            CCG27     104     A
CFA              SX6    X3                                                          CCG27     105     A
CFA              SA6    X2+B3       RND(I) = FLN[BIT(J)]  */ SAVE LINE NUMBER       CCG26      69     A
CFA              SB3    B3+B1       I = I + 1                                       CCG26      70     A
CFA       MRB9   SB4    B4+2        J = J + 2                                       CCG26      71     A
CFA              LT     B4,B5,MRB8  IF J < L.BIT                                    CCG26      72     A
CFA                                                                                CCG26      73     A
CFA              SX6    B3                                                          CCG26      74     A
CFA              SA6    L.RND       L.RND = I                                       CCG26      75     A
CFA              RJ     =YHR$LDC    CALL HOST ROUTINE                               CCG26      76     A
CFA              SX6    0                                                           CCG26      77     A
CFA              SA6    L.RND       L.RND = 0                                       CCG26      78     A
CFA              EQ     MRB                                                         CCG26      79     A
CFA       FGS    TITLE  FGS - FORM GRAPH STRUCTURE TABLES                           CFA       364     A
CFA      **      FGS - FORM GRAPH STRUCTURE TABLES                                  CFA       365     A
CFA      *       FORM EDGE INDEX AND COMPRESS EDGE TABLES FROM *CFT* .              CFA       366     A
CFA                                                                                CFA       367     A
CFA       FGS    ROUTINE                                                            CFA       368     A
CFA              SX1    B1                                                          CFA       369     A
CFA              LX1    CF.FROMP                                                    CFA       370     A
CFA              ADDWRD CFT,X1      FLOW( 1 , 0 ) */ DUMMY EXIT EDGE                CFA       371     A
CFA              ADDWRD CFT,B0      ADDWRD( CFT , 0 )                               CFA       372     A
CFA              SX1    X3-1                                                        CFA       373     A
CFA              SB7    X2                                                          CFA       374     A
CFA              MX0    60-CF.JPP                                                   CFA       375     A
CFA              BX6    X3                                                          CFA       376     A
CFA              SB6    B0                                                          CFA       377     A
CFA              BX0    -X0                                                         CFA       378     A
CFA              MX5    0                                                           CFA       379     A
CFA              SA6    LC          LC = L.CFT    */ SAVE IN CASE GRAPH REDONE      CFA       380     A
CFA              CALL   SST         SORT( CFT , [FROM,TO] )                         CFA       381          I
         -CCG1
CFA              CALL   SST=        SORT( CFT , [FROM,TO] )                         CCG1      372          I
         -CCGNC
CFA              CALL   SST#        SORT( CFT , [FROM,TO] )                         CCGNC     828     A
CFA              TRACE  FGS,CFT                                                     CFA       382     A
CFA                                                                                CFA       383     A
CFA      *       ALLOCATE SPACE FOR EDGE INDEX AND EDGE TABLES                      CFA       384     A
CFA                                                                                CFA       385     A
CFA              SA5    N.BLK                                                       CFA       386     A
CFA              SA3    L.CFT                                                       CFA       387     A
CFA              IX6    X5+X3                                                       CFA       388     A
CFA              SX1    X6+3                                                        CFA       389     A
CFA              ALLOC  GST,X1      ALLOC( GST , 3+N.BLK+L.CFT )                    CFA       390     A
CFA              SB5    B6+2                                                        CFA       391     A
CFA              SX6    B5+X2       EIB = O.GST + OLD(L.GST) + 2                    CFA       392     A
CFA              SX7    B6                                                          CFA       393     A
CFA              SA7    LB+1        LB(2) = OLD(L.GST)  */ BASE OF THIS GRAPH       CFA       394     A
CFA              SA6    EIB                                                         CFA       395     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA                SX4    X5+B1                                          CFA          396     A
CFA                IX7    X6+X4       ETB = EIB + 1 + N.BLK              CFA          397     A
CFA                SA7    A6+B1                                          CFA          398     A
CFA                                                                      CFA          399     A
CFA      *         SCAN *CFT* AND SET SUCCESSOR INDEX INFO IN *GST*      CFA          400     A
CFA      *         NOTE THAT EVERY BLOCK, EXCEPT THE EXIT BLOCK HAS A SUCCESSOR,  CFA  401     A
CFA      *         EXCEPT LOOPS THAT HAVE NO EXITS.                      CFA          402     A
CFA                                                                      CFA          403     A
CFA                SA1    O.CFT                                          CFA          404     A
CFA                SB2    X6          E0 = EIB                           CFA          405     A
CFA                SB4    X7          EF = ETB                           CFA          406     A
CFA                SB7    X1          CF = [O.CFT]                       CFA          407     A
CFA                SX6    2000B                                          CFA          408     A
CFA                SA5    X1          CI = [O.CFT]                       CFA          409     A
CFA                LX6    EI.NSP                                         CFA          410     A
CFA                MX4    1+CF.TOL                                       CFA          411     A
CFA                LX4    CF.TOL                                         CFA          412     A
CFA                MX1    -CF.FROML                                      CFA          413     A
CFA                SB3    B4-B1                                          CFA          414     A
CFA                                                                      CFA          415     A
CFA       FGS0     SA6    B3          FOR I = ETB-1 TO EIB;  NS[I] = 0   CFA          416     A
CFA                SB3    B3-B1                                          CFA          417     A
CFA                GE     B3,B2,FGS0                                     CFA          418     A
CFA                BX7    X4*X5                                          CFA          419     A
CFA                LX1    CF.FROMP                                       CFA          420     A
CFA                MX0    -CF.JPP                                        CFA          421     A
CFA                SA7    B4          ET = ETB;  [ET] = JPTO[CI]         CFA          422     A
CFA                SX2    2000B                                          CFA          423     A
CFA                LX2    EI.NPP                                         CFA          424     A
CFA                                                                      CFA          425     A
CFA       FGS1     BX3    X5          CL = [CI]                          CFA          426     A
CFA                SA5    A5+B1       CI = CI + 1                        CFA          427     A
CFA                BX7    X3-X5                                          CFA          428     A
CFA                BX6    -X0*X7                                         CFA          429     A
CFA                ZR     X6,FGS1     IF [CI] = CL                       CFA          430         I
          -FCC0448
CFA                                                                      CFA          431         I
          -FCC0448
CFA                BX7    X4*X5                                          CFA          432         I
          -FCC0448
CFA                NZ     X6,FGS1A    IF FROMTO[CI] " FROMTO[CL]         FCC0448      189     A
CFA                                                                      FCC0448      190     A
CFA                SA3    A7                                             FCC0448      191     A
CFA                BX6    X0*X5                                          FCC0448      192     A
CFA                BX7    X6+X3       [ET] = [ET] ! JP[CI]  */ OR *JP* BIT  FCC0448   193     A
CFA                SA7    A7                                             FCC0448      194     A
CFA                EQ     FGS1                                           FCC0448      195     A
CFA                                                                      FCC0448      196     A
CFA       FGS1A    BX7    X4*X5                                          FCC0448      197     A
CFA                SA7    A7+B1       ET = ET + 1;  [ET] = JPTO[CI]      CFA          433     A
CFA                BX6    -X1*X6                                         CFA          434     A
CFA                ZR     X6,FGS1     IF FROM[CI] = FROM[CL]             CFA          435     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA                                                                    CFA     436     A
CFA             SB3     A7-B4       NS = ET - EF                       CFA     437     A
CFA             SX6     B4-B2       SUI = EF - E0                      CFA     438     A
CFA             PX7     B3,X6                                          CFA     439     A
CFA             LX3     -CF.FROMP   J = FROM[CL]                       CFA     440     A
CFA             SB4     A7          EF = ET                            CFA     441     A
CFA             LX7     EI.SUIP                                        CFA     442     A
CFA             BX6     X2+X7                                          CFA     443     A
CFA             SA6     B2+X3       (NS,SUI)[EI+J] = (NS,SUI)          CFA     444     A
CFA             NZ      X5,FGS1     IF [CI] " 0                        CFA     445     A
CFA                                                                    CFA     446     A
CFA     *       SET INTERVAL LIST POINTERS IN GRAPH HEADER WORD        CFA     447     A
CFA                                                                    CFA     448     A
CFA             SA4     O.GST                                          CFA     449     A
CFA             SA3     N.BLK                                          CFA     450     A
CFA             SB5     X4-1                                           CFA     451     A
CFA             LX3     GB.NBP      NB[GB] = N.BLK                     CFA     452     A
CFA             SX6     A7-B5                                          CFA     453     A
CFA             SA6     L.GST       L.GST = ET + 1 - O.GST             CFA     454     A
CFA             SA6     ILB         ILB = L.GST   */ INTERVAL LIST BASE CFA     455     A
CFA             SX6     A7-B2                                          CFA     456     A
CFA             SX6     X6+B1       ILI[GB] = ET+1 - EIB  */ INTERVAL LIST INDEX CFA 457 A
CFA             LX6     GB.ILIP                                        CFA     458     A
CFA             IX7     X3+X6                                          CFA     459     A
CFA             SA7     B2-B1       GB= EIB - 1                        CFA     460     A
CFA                                                                    CFA     461     A
CFA     *       SWAP FROM, TO FIELDS IN *CFT* AND SORT TABLE           CFA     462     A
CFA                                                                    CFA     463     A
CFA             MX0     60-CF.JPP                                      CFA     464     A
CFA             SA5     B7          CI = CF                            CFA     465     A
CFA             SB5     CF.FROMP                                       CFA     466     A
CFA     FGS2    BX6     -X0*X5      SWAP( FROM[CI] , TO[CI] )          CFA     467     A
CFA             SA5     A5+B1       CI = CI + 1                        CFA     468     A
CFA             LX7     B5,X6                                          CFA     469     A
CFA             SA7     A5-B1                                          CFA     470     A
CFA             NZ      X5,FGS2     IF [CI] " 0                        CFA     471     A
CFA                                                                    CFA     472     A
CFA             SX1     A5-B7       L.CFT = CI - CF  */ FOR SORT       CFA     473     A
CFA             CALL    SHL         SORT( CFT , [FROM,TO] )            CFA     474         I
        -CCG1
CFA             CALL    SHL=        SORT( CFT , [FROM,TO] )            CCG1    373         I
        -CCGNC
CFA             CALL    SHL#        SORT( CFT , [FROM,TO] )            CCGNC   829     A
CFA                                                                    CFA     475     A
CFA     *       ADD THE PREDECESSOR INFORMATION TO THE EDGE TABLES     CFA     476     A
CFA                                                                    CFA     477     A
CFA             SA1     O.CFT                                          CFA     478     A
CFA             SA3     EIB                                            CFA     479     A
CFA             SA2     A3+B1                                          CFA     480     A
CFA             MX0     -CF.TOL                                        CFA     481     A
CFA             SA5     X1          CI = [O.CFT]                       CFA     482     A
CFA             SA4     X2          ET = [ETB]                         CFA     483     A

        0         1         2         3         4         5         6         7         8
        12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA                SB2    X3          E0 = EIB                                CFA        484    A
CFA                SB4    A4          EF = ET                                 CFA        485    A
CFA                BX2    -X0*X5                                              CFA        486    A
CFA                LX2    ET.PREDP                                            CFA        487    A
CFA                MX1    -CF.FROML                                           CFA        488    A
CFA                BX6    X2+X4       PRED[ET] = TO[CI]                       CFA        489    A
CFA                LX1    CF.FROMP                                            CFA        490    A
CFA                SA6    A4                                                  CFA        491    A
CFA                                                                           CFA        492    A
CFA       FGS3     BX3    X5          CL = [CI]                               CFA        493    A
CFA                SA5    A5+B1       CI = CI + 1                             CFA        494    A
CFA                BX7    X3-X5                                               CFA        495    A
CFA                ZR     X7,FGS3     IF [CI] = CL                            CFA        496    A
CFA                SA4    A4+B1       ET = ET + 1                             CFA        497    A
CFA                BX2    -X0*X5                                              CFA        498    A
CFA                LX2    ET.PREDP                                            CFA        499    A
CFA                IX6    X2+X4                                               CFA        500    A
CFA                SA6    A4          PRED[ET] = TO[CI]                       CFA        501    A
CFA                BX2    -X1*X7                                              CFA        502    A
CFA                ZR     X2,FGS3     IF FROM[CI] = FROM[CL]                  CFA        503    A
CFA                                                                           CFA        504    A
CFA                LX3    -CF.FROMP                                           CFA        505    A
CFA                SB5    A4-B4       NP = ET - EF                            CFA        506    A
CFA                SX6    B4-B2       PRI = EF - E0                           CFA        507    A
CFA                SA2    B2+X3       EI = EIB + FROM[CL]                     CFA        508    A
CFA                PX6    B5,X6                                               CFA        509    A
CFA                LX6    EI.PRIP                                             CFA        510    A
CFA                SB4    A4          EF = ET                                 CFA        511    A
CFA                BX7    X2+X6                                               CFA        512    A
CFA                SA7    A2          NP[EI] = NP;  PRI[EI] = PRI             CFA        513    A
CFA                NZ     X5,FGS3     IF [CI] " 0                             CFA        514    A
CFA                                                                           CFA        515    A
CFA       *        ADJUST GRAPH TABLES TO REMOVE DUMMY EXIT EDGE INFO          CFA        516    A
CFA                                                                           CFA        517    A
CFA                SA2    B2+B1       EI1 = EIT(1)                            CFA        518    A
CFA                SA1    B2          EI0 = EIT(0)                            CFA        519    A
CFA                SX7    B1                                                  CFA        520    A
CFA                LX7    EI.NSP                                              CFA        521    A
CFA                IX2    X2-X7       NS[EI1] = NS[EI1] - 1                   CFA        522    A
CFA                LX7    EI.SUIP-EI.NSP                                      CFA        523    A
CFA                IX6    X2+X7       SUI[EI1] = SUI[EI1] + 1                 CFA        524    A
CFA                LX7    EI.NPP-EI.SUIP                                      CFA        525    A
CFA                SA6    A2                                                  CFA        526    A
CFA                IX1    X1-X7       NP[EI0] = NP[EI0] - 1                   CFA        527    A
CFA                LX7    EI.PRIP-EI.NPP                                      CFA        528    A
CFA                IX6    X1+X7       PRI[EI0] = PRI[EI0] + 1                 CFA        529    A
CFA                SA6    A1                                                  CFA        530    A
CFA                                                                           CFA        531    A
CFA                SA1    O.GST                                               CFA        532    A
CFA                SX2    B2                                                  CFA        533    A
CFA                IX6    X2-X1       EII = EIB - O.GST  */ MAKE RELATIVE     CFA        534    A
CFA                SA6    EII                                                 CFA        535    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA                                                                    CFA     536     A
CFA                TRACE  FGS.X,GST                                    CFA     537     A
CFA                EQ     FGS                                          CFA     538     A
CFA       FIS      TITLE  FIS - FORM INTERVAL STRUCTURE                CFA     539     A
CFA      **        FIS - FORM INTERVAL STRUCTURE OF A PROGRAM FLOW GRAPH  CFA  540     A
CFA      *                                                            CFA     541     A
CFA      *         THE ALGORITHM FOR INTERVAL FORMATION USED HERE IS A VARIANT OF  CFA  542  A
CFA      *         THE ONE DESCRIBED IN AHO-ULLMAN, VOLUME II, PAGES 937 - 941 .  CFA  543  A
CFA      *         THE BACK DOMINATOR CALCULATION IS DERIVED IN THE     CFA     544     A
CFA      *         ARTICLE BY FRAN ALLEN IN THE JULY 1970 SIGPLAN.      CFA     545     A
CFA                                                                    CFA     546     A
CFA      *         B-REGISTER ASSIGNMENT FOR INTERVAL FORMATION/ BACK DOMINATOR  CFA  547  A
CFA      *         CALCULATION LOOPS                                    CFA     548     A
CFA                                                                    CFA     549     A
CFA      *         B2, B3 - SCRATCH, LOOP COUNTS                        CFA     550     A
CFA       IB       MICRO  1,,/B4/     IBA - INTERVAL BASE ADDRESS       CFA     551     A
CFA       GS       MICRO  1,,/B5/     O.GST                             CFA     552     A
CFA       PC       MICRO  1,,/B6/     PC - TEMP SCRATCH TABLE           CFA     553     A
CFA       EI       MICRO  1,,/B7/     EIB - FWA OF EDGE INDEX TABLE     CFA     554     A
CFA                                                                    CFA     555     A
CFA      *         FOR BACK DOMINATOR CALC B4, B7 AS ABOVE AND B5, B6  ARE BV , T  CFA  556  A
CFA       BV       MICRO  1,,/B5/     BV - O.RND , BIT VECTOR TABLE     CFA     557     A
CFA       T        MICRO  1,,/B6/     T - TEMP SCRATCH TABLE            CFA     558     A
CFA                                                                    CFA     559     A
CFA       FIS      ROUTINE                                             CFA     560     A
CFA                SA5    N.BLK                                       CFA     561     A
CFA                SX0    X5+2                                        CFA     562     A
CFA                BX1    X0                                          CFA     563     A
CFA                LX1    2                                           CFA     564     A
CFA                ALLOC  GST,X1      ALLOC( GST , 4*(N.BLK+2) )      CFA     565     A
CFA                SA4    EII                                         CFA     566     A
CFA                SB2    X5          N = N.BLK                       CFA     567     A
CFA                S"GS"  X2                                          CFA     568     A
CFA                S"EI"  "GS"+X4     EIB = O.GST + EII                CFA     569     A
CFA                IX3    X2+X3                                       CFA     570     A
CFA                IX6    X3-X0       HTL = O.GST + L.GST - ( N.BLK + 2 )  CFA  571  A
CFA                S"PC"  X6+B1       O.PCT = HTL + 1                  CFA     572     A
CFA                SA6    HTL                                         CFA     573     A
CFA                                                                    CFA     574     A
CFA      *         SETUP HEADER NODE LIST AND PRED COUNT TABLE         CFA     575     A
CFA                                                                    CFA     576     A
CFA                SX7    B1                                          CFA     577     A
CFA                SA7    X6          HI = HTL;  [HI] = 1              CFA     578     A
CFA                SA0    A7-B1       HI = HI - 1                      CFA     579     A
CFA                MX7    0                                           CFA     580         I
          -FCC0392
CFA                SA7    LINT        LINT = 0      */ LAST INTERVAL   CFA     581         I
          -FCC0392
CFA                                                                    CFA     582     A
CFA                SA5    "EI"+B1     EI = EIB + 1                     CFA     583     A
CFA                SA4    FISB                                        CFA     584     A
CFA                BX6    X4                                          CFA     585     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA              SA6    "PC"        PI = O.PCT;  INI[PI] = 1 */ EXIT NODE *INI*        CFA          586     A
CFA              SX0    B1                                                             FCC1191        8     A
CFA              LX0    I.1PP                                                          FCC1191        9     A
CFA                                                                                    CFA          587     A
CFA       FIS1   LX5    -EI.PRIP                                                       CFA          588     A
CFA              UX4    B3,X5                                                          CFA          589     A
CFA              SX6    B3                                                             CFA          590     A
CFA              LX6    I.PCP                                                          CFA          591     A
CFA              SA6    A6+B1       PI = PI + 1;  PC[PI] = NP[EI]                      CFA          592         I
          -FCC1191
CFA              NE     B3,B1,FIS1A IF NP[EI] " 1                                      FCC1191       10     A
CFA              BX6    X0+X6       1P[P2+1] = 1                                       FCC1191       11     A
CFA       FIS1A  SA6    A6+B1       PI = PI + 1]  PC[PI] = NP[EI]                      FCC1191       12     A
CFA              SB2    B2-B1       N = N - 1                                          CFA          593     A
CFA              SA5    A5+B1       EI = EI + 1                                        CFA          594     A
CFA              NZ     B2,FIS1     IF N " 0                                           CFA          595     A
CFA       FIS    SPACE  2,8                                                            CFA          596     A
CFA      *       INTERVAL FORMATION LOOP                                               CFA          597     A
CFA      *       SELECT NEXT HEADER NODE AS MIN OF NODES IN HEADER LIST NOT IN         CFA          598     A
CFA      *       AN INTERVAL, TO AVOID BLOCK NUMBER REVERSALS AND EXCESSIVE            CFA          599     A
CFA      *       RANDOM READS LATER ON IN *GPO*.                                       CFA          600     A
CFA                                                                                    CFA          601     A
CFA              SA1    ILB                                                            CFA          602     A
CFA              S"IB"  "GS"+X1     IBA = O.GST + ILB  */ INTERVAL BASE ADDRESS        CFA          603     A
CFA              TRACE  FIS1,GST                                                       CFA          604     A
CFA                                                                                    CFA          605     A
CFA       FIS2   SA3    HTL                                                            CFA          606     A
CFA              MX7    0                                                              CFA          607     A
CFA              SA7    A0          [HI] = 0      */ SEARCH TERMINATOR                 CFA          608     A
CFA              SB3    377777B     MHN = *INFINITY*  */ MIN *HN*                      CFA          609     A
CFA              MX0    0           HNA = 0       */ HEADER NODE ADDRESS               CFA          610     A
CFA              SA5    X3+B1       J = HTL + 1                                        CFA          611     A
CFA                                                                                    CFA          612     A
CFA       FIS3   SA5    A5-B1       J = J - 1;  HN = [J]                               CFA          613     A
CFA              SB2    X5                                                             CFA          614     A
CFA              ZR     X5,FIS4     IF HN = 0                                          CFA          615     A
CFA              GE     B2,B3,FIS3  IF HN > MHN                                        CFA          616     A
CFA              SA4    "PC"+X5                                                        CFA          617     A
CFA              LX4    59-I.INIP                                                      CFA          618     A
CFA              MI     X4,FIS3     IF INI[PCT(HN)]                                    CFA          619     A
CFA              SX0    A5          HNA = J                                            CFA          620     A
CFA              SB3    B2          MHN = HN                                           CFA          621     A
CFA              EQ     FIS3                                                           CFA          622     A
CFA                                                                                    CFA          623     A
CFA       FIS4   ZR     X0,FIS15    IF HNA = 0    */ NOTHING FOUND                     CFA          624     A
CFA              SA5    X0          HN = [HLA]                                         CFA          625     A
CFA              SA1    A0+B1                                                          CFA          626     A
CFA              BX6    X1                                                             CFA          627     A
CFA              SA6    A5          SWAP( [HI+1] , [HLA] )                             CFA          628     A
CFA              BX7    X5                                                             CFA          629     A
CFA              SA7    A1                                                             CFA          630     A
CFA              SA0    A0+B1       HI = HI + 1                                        CFA          631     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA                SA4    "PC"+X5     PCW = PCT(HN)                                  CFA          632     A
CFA                                                                                   CFA          633     A
CFA      *         NOW REMOVE ANY ENTRIES THAT ARE *INI*                              CFA          634     A
CFA                                                                                   CFA          635     A
CFA       FIS4A    SA2    X3          J = [HTL]                                       CFA          636     A
CFA                SA1    "PC"+X2                                                     CFA          637     A
CFA                SX3    X3-1        HTL = HTL - 1                                   CFA          638     A
CFA                LX1    59-I.INIP                                                   CFA          639     A
CFA                MI     X1,FIS4A    IF INI[PCT(J)]                                  CFA          640     A
CFA                SX7    X3+B1                                                       CFA          641     A
CFA                SA7    A3          HTL = HTL + 1                                   CFA          642     A
CFA                                                                                   FCC0408       93     A
CFA                MX6    0                                                           FCC0408       94     A
CFA                SA6    FTH         FTH = 0                                         FCC0408       95     A
CFA       FIS      SPACE  3,14                                                        CFA          643     A
CFA      *         PLACE HEADER NODE IN INTERVAL                                      CFA          644     A
CFA                                                                                   CFA          645     A
CFA                MX1    -ET.JPP                                                     CFA          646     A
CFA                BX6    X1*X5                                                       CFA          647     A
CFA                SA6    "IB"        [IBA] = JP[HN]  */ SAVE *JP* INDICATOR          CFA          648     A
CFA                SX5    X5                                                          CFA          649     A
CFA                LX4    -I.PCP                                                      CFA          650     A
CFA                SX6    X4                                                          CFA          651     A
CFA                SA2    FISA                                                        CFA          652     A
CFA                BX7    X2+X5                                                       CFA          653     A
CFA                SA7    "IB"+B1     II = IBA + 1;  [II] = INTM(AP,HN,0,0,HN)        CFA          654     A
CFA                SA6    NPH         NPH = PC[PCW] */ N.PRED TO HN ON ENTRY          CFA          655     A
CFA                                                                                   CFA          656     A
CFA                LX4    I.PCP                                                       CFA          657     A
CFA                SX0    B1                                                          CFA          658     A
CFA                LX0    I.PCP                                                       CFA          659     A
CFA                LX5    I.HNP                                                       CFA          660     A
CFA                MX3    1                                                           CFA          661     A
CFA                LX3    1+I.INIP                                                    CFA          662     A
CFA                BX1    X3+X5                                                       CFA          663     A
CFA                LX3    I.INHP-I.INIP                                               CFA          664     A
CFA                BX2    X3+X5                                                       CFA          665     A
CFA                SX7    A7                                                          CFA          666     A
CFA                LX7    I.IAP                                                       CFA          667     A
CFA                LX6    I.PCP                                                       CFA          668     A
CFA                IX3    X6+X7                                                       CFA          669     A
CFA                BX6    X3+X1       PCT(HN) = ITW(INI,0,HN,II,NPH)                  CFA          670     A
CFA                SA6    A4                                                          CFA          671     A
CFA                                                                                   CFA          672     A
CFA      *         SCAN THE ELEMENTS IN THE INTERVAL AND DECREMENT THE PRED COUNT     CFA          673     A
CFA      *         OF THEIR SUCCESSORS, AND ADD NODES TO INTERVAL IF ALL PRED         CFA          674     A
CFA      *         ARE IN IT, AND NODE WAS FIRST REACHED IN THIS INTERVAL.            CFA          675     A
CFA      *         (A5) = ICN, FETCH ADDR  , (A7) = II, STORE ADDR                    CFA          676     A
CFA      *         (X0) = 1, (X1) = *INI*+BN[HN], (X2) = *INH*+BN[HN]                 CFA          677     A
CFA                                                                                   CFA          678     A
CFA                SA5    A7          ICN = II      */ INDEX TO CURRENT NODE          CFA          679     A
CFA                BX7    -X5                                                         CFA          680     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA                SB2    X7          HN = BN[II]                                  CFA          681     A
CFA                                                                                CFA          682     A
CFA       FIS5     LX5    -IM.BNP                                                   CFA          683     A
CFA                SA4    "EI"+X5     EI = EIB + BN[ICN]                           CFA          684     A
CFA                LX4    -EI.SUIP                                                  CFA          685     A
CFA                UX7    B3,X4       N = NS[EI]                                   CFA          686     A
CFA                SA4    "EI"+X4     SI = EIB + SUI[EI]                           CFA          687     A
CFA                LE     B3,FIS10A   IF N @ 0                                     CFA          688     A
CFA                                                                                CFA          689     A
CFA       FIS6     LX4    -ET.SUCCP   I = SUCC[SI]                                 CFA          690     A
CFA                SA3    "PC"+X4     PW = PCT(I)                                  CFA          691     A
CFA                IX6    X3-X0       PC[PW] = PC[PW] - 1                          CFA          692     A
CFA                LX3    59-I.INIP                                                CFA          693     A
CFA                LX7    B1,X3                                                    CFA          694     A
CFA                SA6    A3                                                       CFA          695     A
CFA                MI     X3,FIS9     IF INI[PW]                                   CFA          696     A
CFA                MI     X7,FIS7     IF INH[PW]                                   CFA          697     A
CFA                                                                                CFA          698     A
CFA      *         CHECK ADD TO HEADER LIST                                        CFA          699     A
CFA                                                                                CFA          700     A
CFA                SX7    X6                                                       CFA          701     A
CFA                ZR     X7,FIS8     IF PC[PW] = 0                                CFA          702     A
CFA                BX6    X2+X6       INH[PW] = 1;  HN[PW] = HN                    CFA          703     A
CFA                SA6    A6                                                       CFA          704     A
CFA                SX6    X4                                                       CFA          705     A
CFA                LX4    ET.SUCCP+59-ET.JPP                                       CFA          706     A
CFA                MX7    1                                                        CFA          707     A
CFA                BX4    X7*X4                                                    CFA          708     A
CFA                BX6    X4+X6       [HI] = JP[SI] ! I                            CFA          709     A
CFA                SA6    A0                                                       CFA          710     A
CFA                SA0    A0-B1       HI = HI - 1                                  CFA          711     A
CFA                EQ     FIS10                                                    CFA          712     A
CFA                                                                                CFA          713     A
CFA      *         IN HEADER LIST, CHECK ADD TO INTERVAL                           CFA          714     A
CFA                                                                                CFA          715     A
CFA       FIS7     SX7    X6                                                       CFA          716     A
CFA                BX3    X6-X2                                                    CFA          717     A
CFA                NZ     X7,FIS10    IF PC[PW] " 0                                CFA          718     A
CFA                LX3    -I.HNP                                                   CFA          719     A
CFA                SX7    X3                                                       CFA          720     A
CFA                NZ     X7,FIS10    IF HN[PW] " HN */ FIRST REACHED BY ANOTHER   CFA          721     A
CFA                                                                                CFA          722     A
CFA      *         ADD NODE TO THIS INTERVAL                                       CFA          723     A
CFA                                                                                CFA          724     A
CFA       FIS8     SX7    X4                                                       CFA          725     A
CFA                SA7    A7+B1       II = II + 1;  [II] = I                       CFA          726     A
CFA                SX3    A7                                                       CFA          727     A
CFA                LX3    I.IAP                                                    CFA          728     A
CFA                SX7    X6                        */ PRESERVE PRED COUNT         FCC1191       13     A
CFA                BX3    X3+X7                                                    FCC1191       14     A
CFA                BX6    X3+X1       INI[PW] = 1;  IA[PW] = II                    CFA          729     A
CFA                SA6    A6                                                       CFA          730           I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

          -FCC1191
CFA                 SA6    A3                                                   FCC1191      15     A
CFA                                                                             FCC1191      16     A
CFA       *         CHECK IMMEDIATE SUCC AND ADD IT TO THE INTERVAL IF IT        FCC1191      17     A
CFA       *         IS THE ONLY EDGE BETWEEN THE ADDED NODE AND THE SUCCESSOR.   FCC1191      18     A
CFA                                                                             FCC1191      19     A
CFA                 SA3    "EI"+X4     EI = EIB + I                             FCC1191      20     A
CFA                 LX3    -EI.SUIP                                             FCC1191      21     A
CFA                 UX7    B1,X3                                                FCC1191      22     A
CFA                 SX6    B1-1                                                 FCC1191      23     A
CFA                 SB1    1                                                    FCC1191      24     A
CFA                 NZ     X6,FIS10    IF NS[EI] > 1                            FCC1191      25     A
CFA                 SA3    "EI"+X3     SIS = EIB + SUI[EI]                      FCC1191      26     A
CFA                 LX3    -ET.SUCCP                                            FCC1191      27     A
CFA                 SX4    X3          I = SUCC[SIS]                            FCC1191      28     A
CFA                 SA3    "PC"+X4     PW = PCT(I)                              FCC1191      29     A
CFA                 LX3    59-I.1PP                                             FCC1191      30     A
CFA                 SX6    B1          PC[PW] = 1                               FCC1191      31     A
CFA                 MI     X3,FIS8     IF 1P[PW]     */ ONLY 1 PRED             FCC1191      32     A
CFA                 EQ     FIS10                                                CFA         731     A
CFA                                                                             CFA         732     A
CFA        FIS9     SX3    B2+X4                                                CFA         733     A
CFA                 NZ     X3,FIS10    IF SUCC[SI] " HN                         CFA         734     A
CFA                                                                             CFA         735     A
CFA                 LX0    IM.LNP                                               CFA         736     A
CFA                 BX6    X0+X5       LN[ICN] = 1   */ INDICATE A LATCHING NODE CFA         737     A
CFA                 LX0    -IM.LNP                                              CFA         738         I
          -FCC0408
CFA                 SA6    A5+                                                  CFA         739         I
          -FCC0408
CFA                 SA6    A5                                                   FCC0408      96     A
CFA                 SX0    B1                                                   FCC0408      97     A
CFA                 SX6    B1                                                   FCC0408      98     A
CFA                 LX4    59-ET.JPP+ET.SUCCP                                   FCC0408      99     A
CFA                 MI     X4,FIS10    IF JP[SI]     */ REACHED BY A JUMP       FCC0408     100     A
CFA                                                                             FCC0408     101     A
CFA                 SA6    FTH         FTH = 1       */ INDICATE FALL THROUGH TO H FCC0408     102     A
CFA                                                                             CFA         740     A
CFA        FIS10    SB3    B3-B1       N = N - 1                                CFA         741     A
CFA                 SA4    A4+B1       SI = SI + 1                              CFA         742     A
CFA                 GT     B3,FIS6     IF N > 0                                 CFA         743     A
CFA                                                                             CFA         744     A
CFA       *         CONTINUE THIS INTERVAL IF ELEMENTS WERE ADDED                CFA         745     A
CFA                                                                             CFA         746     A
CFA        FIS10A   SB3    A5                                                   CFA         747     A
CFA                 SX6    A7-B3                                                CFA         748     A
CFA                 ZR     X6,FIS11    IF II = ICN   */ END OF INTERVAL         CFA         749     A
CFA                                                                             CFA         750     A
CFA                 SA5    A5+B1       ICN = ICN + 1                            CFA         751     A
CFA                 EQ     FIS5                                                 CFA         752     A
CFA        FIS      SPACE  2,14                                                 CFA         753     A
CFA       *         TERMINATE THIS INTERVAL, SETUP HEADER WORD                   CFA         754     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA                                                                                  CFA          755     A
CFA      FIS11    SA3    LINT                                                        CFA          756          I
        -FCC0392
CFA               SX1    A5-"IB"     NODE[IBA] = ICN - IBA  */ N.NODES IN INTERVAL   CFA          757          I
        -FCC0392
CFA               LX1    IP.NODEP                                                    CFA          758          I
        -FCC0392
CFA               LX3    IP.LBP      LB[IBA] = LINT                                  CFA          759          I
        -FCC0392
CFA               IX7    X1+X3                                                       CFA          760          I
        -FCC0392
CFA               LX1    -IP.NODEP                                                   CFA          761          I
        -FCC0392
CFA               SX6    X1+B1       LINT = NODE[IBA] + 1  */ NEW LINK BACKWARDS     CFA          762          I
        -FCC0392
CFA               SA6    A3                                                          CFA          763          I
        -FCC0392
CFA                                                                                  CFA          764          I
        -FCC0392
CFA               SA1    "IB"+B1     IHA = IBA + 1                                   CFA          765          I
        -FCC0392
CFA      FIS11    SA1    "IB"+B1     IHA = IBA + 1                                   FCC0392      144     A
CFA               SA2    NPH                                                         CFA          766     A
CFA               LX1    -IM.BNP     HN = BN[IHA]  */ HEADER NODE                    CFA          767     A
CFA               SA3    "PC"+X1                                                     CFA          768     A
CFA               LX3    -I.PCP                                                      CFA          769     A
CFA               SX5    X3                                                          CFA          770     A
CFA               IX0    X2-X5       N.LN = NPH - PC[ PCT(HN) ]                      CFA          771     A
CFA               SA4    "EI"+X1     EI = EIB + HN                                   CFA          772     A
CFA               LX1    IM.BNP                                                      CFA          773     A
CFA               LX4    -EI.PRIP                                                    CFA          774     A
CFA               SA2    "IB"        IB0 = IBA;  GLN = [IBA] */ GL NEEDED FLAG       CFA          775     A
CFA               UX4    B3,X4                                                       CFA          776     A
CFA               SX5    B3                                                          CFA          777     A
CFA               IX3    X5-X0       N.XP = NP[EI] - N.LN                            CFA          778     A
CFA               SB2    X3                                                          CFA          779     A
CFA               SX7    A5-"IB"     NODE[IBA] = ICN - IBA */ N.NODES IN INTERVA     FCC0392      145     A
CFA               SX6    X7+B1                                                       FCC0392      146     A
CFA               LX7    IP.NODEP                                                    FCC0392      147     A
CFA               S"IB"  "IB"+X6     IBA = IBA + NODE[IBA] + 1  */ ADVANCE *IBA*     CFA          780     A
CFA               EQ     B2,B1,FIS11A      IF N.XP = 1                               CFA          781     A
CFA               MX2    1           GLN = 1                                         CFA          782     A
CFA      FIS11A   LX2    1+IP.GLP                                                    CFA          783     A
CFA               BX7    X2+X7       GL[IB0] = GLN                                   CFA          784     A
CFA               LX0    IP.NIPP                                                     FCC0392      148     A
CFA               BX7    X0+X7       NIP[IB0] = N.LN  */ N.INTERNAL PRED             FCC0392      149     A
CFA               SA7    A2                                                          CFA          785     A
CFA               ZR     X0,FIS2     IF N.LN = 0   */ NO LATCHING NODES              CFA          786     A
CFA                                                                                  FCC0408      103     A
CFA               SA3    L.LCT                                                       FCC1319      137          I
        -CCG18
CFA               SA3    L.LCT#                                                      CCG18        182     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA                 LX2     -IP.GLP                                                 FCC1319     138     A
CFA                 IX6     X2+X3       L.LCT = L.LCT + GLN                         FCC1319     139     A
CFA                 SA6     A3                                                      FCC1319     140     A
CFA                 SA3     FTH                                                     FCC0408     104     A
CFA                 SA2     =XN.FTH#                                                CCGA021      63     A
CFA                 IX6     X2+X3       N.FTH = N.FTH + FTH                         CCGA021      64     A
CFA                 SA6     A2                                                      CCGA021      65     A
CFA                 LX3     IP.FTP                                                  FCC0408     105     A
CFA                 BX7     X3+X7       FT[IB0] = FTH                               FCC0408     106     A
CFA                 SA2     CBN         PB[IB0] = CBN */ SET PROLOGUE BLOCK NUMBER  CFA         787         I
            -CCG1
CFA                 SA2     CG.CBN      PB[IB0] = CBN */ SET PROLOGUE BLOCK NUMBER  CCG1        374         I
            -CCGNC
CFA                 SA2     CC$CBN      PB[IB0] = CBN */ SET PROLOGUE BLOCK NUMBER  CCGNC       830     A
CFA                 SX6     X2+B1       CBN = CBN + 1                               CFA         788     A
CFA                 LX2     IP.PBP                                                  CFA         789     A
CFA                 SA6     A2                                                      CFA         790     A
CFA                 BX7     X2+X7                                                   CFA         791     A
CFA                 SA7     A7                                                      CFA         792     A
CFA                                                                                 CFA         793     A
CFA         *       MARK NODES THAT ARE IN *SCR* OF INTERVAL                        CFA         794     A
CFA         *       BY MARKING ALL PRED OF THE LN AS *SCR*                          CFA         795     A
CFA                                                                                 CFA         796     A
CFA                 SA5     A5+B1       II = ICN + 1                                CFA         797     A
CFA                 MX0     1                                                       CFA         798     A
CFA                 LX0     1+IM.SCRP                                               CFA         799     A
CFA                 SB2     59-IM.LNP                                               CFA         800     A
CFA                                                                                 CFA         801     A
CFA         FIS12   SA5     A5-B1       II = II - 1                                 CFA         802     A
CFA                 LX6     B2,X5                                                   CFA         803     A
CFA                 LX5     59-IM.SCRP                                              CFA         804     A
CFA                 BX3     X6+X5                                                   CFA         805     A
CFA                 PL      X3,FIS12    IF ^(LN[II] ! SCR[II])                      CFA         806     A
CFA                 LX5     1+IM.SCRP                                               CFA         807     A
CFA                 BX7     X0+X5       SCR[II] = 1   */ MARK NODE AS IN SCR        CFA         808     A
CFA                 LX6     IM.LNP-IM.HNP                                           CFA         809     A
CFA                 SA7     A5                                                      CFA         810     A
CFA                 MI      X6,FIS2     IF HN[II]     */ END OF INTERVAL            CFA         811     A
CFA                                                                                 CFA         812     A
CFA                 LX5     -IM.BNP                                                 CFA         813     A
CFA                 SA4     "EI"+X5     EI = EIB + BN[II]                           CFA         814     A
CFA                 LX4     -EI.PRIP                                                CFA         815     A
CFA                 UX6     B3,X4       N = NP[EI]                                  CFA         816     A
CFA                 SA3     "EI"+X4     PI = EIB + PRI[EI]                          CFA         817     A
CFA                                                                                 CFA         818     A
CFA         FIS13   LX3     -ET.PREDP                                               CFA         819     A
CFA                 SA2     "PC"+X3                                                 CFA         820     A
CFA                 LX2     -I.IAP      NA = IA[PCT(PRED[PI])]                      CFA         821     A
CFA                 SA1     X2                                                      CFA         822     A
CFA                 BX6     X0+X1       SCR[NA] = 1                                 CFA         823     A
CFA                 SA6     A1                                                      CFA         824     A
CFA                 SB3     B3-B1       N = N - 1                                   CFA         825     A

            0         1         2         3         4         5         6         7         8
            12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA              SA3    A3+B1       PI = PI + 1                                  CFA          826     A
CFA              NZ     B3,FIS13    IF N " 0                                     CFA          827     A
CFA                                                                             CFA          828     A
CFA              EQ     FIS12                                                    CFA          829     A
CFA       FIS    SPACE  3                                                        CFA          830     A
CFA      *       TERMINATE INTERVAL LIST                                         CFA          831     A
CFA                                                                             CFA          832     A
CFA       FIS15  MX6    0                                                        CFA          833     A
CFA              SA6    "IB"        [IBA] = 0     */ LIST TERMINATOR             CFA          834     A
CFA              SA2    PBN                                                      CFA          835     A
CFA              SA3    CBN                                                      CFA          836         I
         -CCG1
CFA              SA3    CG.CBN                                                   CCG1         375         I
         -CCGNC
CFA              SA3    CC$CBN                                                   CCGNC        831     A
CFA              SX1    "IB"-"GS"                                                CFA          837     A
CFA              SA4    "EI"-B1     GB = EIB - 1                                 CFA          838     A
CFA              IX6    X3-X2       SCR = CBN - PBN */ NUMBER OF LOOPS           CFA          839     A
CFA              SX2    X6                                                       FCC0392      150     A
CFA              SX7    X1+B1       L.GST = IBA + 1 - O.GST                      CFA          840     A
CFA              SA7    L.GST                                                    CFA          841     A
CFA              SA6    SCR                                                      CFA          842     A
CFA              LX6    GB.NLPP                                                  CFA          843     A
CFA              BX6    X6+X4       NLP[GB] = SCR */ N. LOOPS                    CFA          844     A
CFA              SA6    A4                                                       CFA          845     A
CFA              SA1    =XM.LP                                                   FCC0392      151     A
CFA              IX3    X1-X2                                                    FCC0392      152     A
CFA      +       PL     X3,*+1      M.LP = MAX( M.LP , SCR )                     FCC0392      153     A
CFA              BX6    X2                                                       FCC0392      154     A
CFA              SA6    A1                                                       FCC0392      155     A
CFA                                                                             CFA          846     A
CFA      *       SETUP FOWARD/BACKWARD LINKS IN GRAPH                            CFA          847     A
CFA                                                                             CFA          848     A
CFA              SA2    LB                                                       CFA          849     A
CFA              SA3    A2+B1                                                    CFA          850     A
CFA              IX6    X7-X3       LF = L.GST - LB(2)                           CFA          851     A
CFA              LX2    GA.LBP                                                   CFA          852     A
CFA              LX6    GA.LFP                                                   CFA          853     A
CFA              BX7    X6+X2                                                    CFA          854     A
CFA              SA7    "EI"-2      [EIB-2] = GA(LB,LF)                          CFA          855     A
CFA              SA4    M.GRL                                                    CFA          856     A
CFA      +       IX1    X6-X4                                                    CFA          857     A
CFA              MI     X1,*+1      M.GRL = MAX( M.GRL , LF )                    CFA          858     A
CFA              SA6    A4                                                       CFA          859     A
CFA       FIS    SPACE  3,14                                                     CFA          860     A
CFA      *       MARK THE EXIT NODES OF THE INTERVALS. THEY ARE THE PRED OF THE  CFA          861     A
CFA      *       INTERVAL HEADER NODES.                                          CFA          862     A
CFA                                                                             CFA          863     A
CFA              SA1    ILB                                                      CFA          864     A
CFA              SX0    B1                                                       CFA          865     A
CFA              LX0    IM.ENP                                                   CFA          866     A
CFA              SB2    -B0         HN = 0        */ START WITH PROG EXIT        CFA          867     A

         0        1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA                SA1    "GS"+X1     NIP = O.GST + ILB  */ NEXT TO PROCESS          CFA        868    A
CFA                                                                                   CFA        869    A
CFA       FIS16    SA5    "EI"-B2     EI = EIB + HN                                   CFA        870    A
CFA                LX5    -EI.PRIP                                                    CFA        871    A
CFA                UX6    B3,X5       N = NP[EI]                                      CFA        872    A
CFA                SA4    "EI"+X5     PI = EIB + PRI[EI]                              CFA        873    A
CFA                LE     B3,FIS19    IF N @ 0                                        CFA        874    A
CFA                                                                                   CFA        875    A
CFA       FIS17    LX4    -ET.PREDP                                                   CFA        876    A
CFA                SA3    "PC"+X4                                                     CFA        877    A
CFA                LX3    -I.IAP                                                      CFA        878    A
CFA                SA2    X3          NA = IA[PCT(PRED[PI])]                          CFA        879    A
CFA                LX3    I.IAP-I.HNP                                                 CFA        880    A
CFA                SX5    X3+B2                                                       CFA        881    A
CFA                ZR     X5,FIS18    IF HN[NA] = HN  */ PRED IN THIS INTERVAL        CFA        882    A
CFA                SX6    A2                                                          CCG16      287    A
CFA                ZR     X6,FIS18    IF NA = 0                                       CCG16      288    A
CFA      *[        CFA                                                                CCG16      289    A
CFA                BX6    X0+X2                                                       CFA        883    A
CFA                SA6    A2          EN[NA] = 1                                      CFA        884    A
CFA                                                                                   CFA        885    A
CFA       FIS18    SB3    B3-B1       N = N - 1                                       CFA        886    A
CFA                SA4    A4+B1       PI = PI + 1                                     CFA        887    A
CFA                NZ     B3,FIS17    IF N " 0                                        CFA        888    A
CFA                                                                                   CFA        889    A
CFA       FIS19    LX1    -IP.NODEP                                                   CFA        890    A
CFA                SB3    X1                                                          CFA        891    A
CFA                ZR     B3,FIS20    IF NODE[NIP] = 0  */ END OF INTERVAL LIST       CFA        892    A
CFA                SA2    A1+B1                                                       CFA        893    A
CFA                LX2    -IM.BNP                                                     CFA        894    A
CFA                BX3    -X2                                                         CFA        895    A
CFA                SB2    X3          HN = BN[NIP+1]                                  CFA        896    A
CFA                SA1    A2+B3       NIP = NIP + NODE[NIP]                           CFA        897    A
CFA                EQ     FIS16                                                       CFA        898    A
CFA                                                                                   CFA        899    A
CFA                                                                                   CFA        900    A
CFA      *         SETUP THE *BI* FIELD IN THE INTERVAL MEMBERS                        CFA        901    A
CFA                                                                                   CFA        902    A
CFA       FIS20    SA1    L.OTI                                                       CFA        903    A
CFA                SA2    ILB                                                         CFA        904    A
CFA                ZR     X1,FIS23    IF L.OTI = 0  */ FIRST GRAPH                    CFA        905    A
CFA                                                                                   CFA        906    A
CFA                SA3    O.OTI                                                       CFA        907    A
CFA                SA5    "GS"+X2     IP = O.GST + ILB                                CFA        908    A
CFA                MX6    0                                                           CFA        909    A
CFA                SB3    X3          T = O.OTI                                       CFA        910    A
CFA                SA6    A1          L.OTI = 0                                       CFA        911    A
CFA                                                                                   CFA        912    A
CFA       FIS21    LX5    -IP.NODEP                                                   CFA        913    A
CFA                SB2    X5+         N = NODE[IP]                                    CFA        914    A
CFA                SA5    A5+1        IP = IP + 1                                     CFA        915    A
CFA                                                                                   CFA        916    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA       FIS22    LX5    -IM.BNP                                          CFA          917     A
CFA                SA4    B3+X5                                            CFA          918     A
CFA                LX5    IM.BNP                                           CFA          919     A
CFA                LX4    IM.BIP+1                                         CFA          920     A
CFA                BX6    X4+X5       BI[IP] = 2*T(BN[IP])                 CFA          921     A
CFA                SA6    A5                                               CFA          922     A
CFA                SB2    B2-B1       N = N - 1                            CFA          923     A
CFA                SA5    A5+B1       IP = IP + 1                          CFA          924     A
CFA                NZ     B2,FIS22    IF N " 0                             CFA          925     A
CFA                NZ     X5,FIS21    IF [IP] " 0                          CFA          926     A
CFA                                                                       CFA          927     A
CFA                EQ     FIS26                                            CFA          928     A
CFA                                                                       CFA          929     A
CFA      *         SET *BI* FIELD FOR *IM* IN FIRST GRAPH                  CFA          930     A
CFA                                                                       CFA          931     A
CFA       FIS23    SA5    "GS"+X2     IP = O.GST + ILB                     CFA          932     A
CFA                SB2    IM.BIP+1                                         CFA          933     A
CFA                                                                       CFA          934     A
CFA       FIS24    LX5    -IP.NODEP                                        CFA          935     A
CFA                SB3    X5+         N = NODE[IP]                         CFA          936     A
CFA                SA5    A5+1        IP = IP + 1                          CFA          937     A
CFA                                                                       CFA          938     A
CFA       FIS25    LX5    -IM.BNP                                          CFA          939     A
CFA                SX4    X5                                               CFA          940     A
CFA                LX5    IM.BNP                                           CFA          941     A
CFA                LX3    B2,X4                                            CFA          942     A
CFA                BX6    X3+X5       BI[IP] = 2*BN[IP]                    CFA          943     A
CFA                SA6    A5                                               CFA          944     A
CFA                SB3    B3-B1       N = N - 1                            CFA          945     A
CFA                SA5    A5+B1       IP = IP + 1                          CFA          946     A
CFA                NZ     B3,FIS25    IF N " 0                             CFA          947     A
CFA                NZ     X5,FIS24    IF [IP] " 0                          CFA          948     A
CFA        FIS     SPACE  3,14                                             CFA          949     A
CFA      *         RESCAN THE INTERVALS AND REORDER THEM SO AS TO GET RID OF ANY  CFA   950     A
CFA      *         REVERSALS WHICH ARE NOT CONSISTANT WITH PROGRAM ORDER.  CFA          951     A
CFA                                                                       CFA          952     A
CFA        FIS26   SA5    "GS"+X2     IP = O.GST + ILB                     CFA          953     A
CFA                                                                       CFA          954     A
CFA        FIS27   LX5    -IP.NODEP                                        CFA          955     A
CFA                SB2    X5-2        N = NODE[IP] - 2                     CFA          956     A
CFA                GT     B2,FIS28    IF N > 2                             CFA          957     A
CFA                SB2    X5+B1                                            CFA          958     A
CFA                SA5    A5+B2       IP = IP + N+1                        CFA          959     A
CFA                NZ     X5,FIS27    IF [IP] " 0                          CFA          960     A
CFA                EQ     CBD                                              CFA          961     A
CFA                                                                       CFA          962     A
CFA      *         SCAN INTERVAL AND CHECK FOR REVERSALS                   CFA          963     A
CFA                                                                       CFA          964     A
CFA        FIS28   SA0    A5          IPF = IP                             CFA          965     A
CFA                SA5    A5+2        IP = IP + 2   */ FIRST AFTER HEADER  CFA          966     A
CFA                SB6    B0          SWAP = 0                             CFA          967     A
CFA                BX7    X5          LIW = [IP]                           CFA          968     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA                LX5    -IM.BNP                                                  CFA        969    A
CFA                SB5    X5                                                       CFA        970    A
CFA                                                                                CFA        971    A
CFA       FIS29    SA5    A5+B1       IP = IP + 1                                  CFA        972    A
CFA                SB2    B2-B1       N = N - 1                                    CFA        973    A
CFA                LX5    -IM.BNP                                                  CFA        974    A
CFA                SB4    X5+                                                      CFA        975    A
CFA                LX5    IM.BNP                                                   CFA        976    A
CFA                GT     B4,B5,FIS31 IF BN[IP] > BN[LIW]  */ IN ORDER             CFA        977    A
CFA                                                                                CFA        978    A
CFA      *         SWAP NODES IF *LIW* IS NOT A PRED OF *IP*                        CFA        979    A
CFA                                                                                CFA        980    A
CFA                SA4    "EI"+X7     EI = EIB + BN[LIW]                           CFA        981    A
CFA                SB4    -B4                                                      CFA        982    A
CFA                LX4    -EI.SUIP                                                 CFA        983    A
CFA                SA3    "EI"+X4     SI = EIB + SUI[EI]                           CFA        984    A
CFA                UX6    B3,X4       NS = NS[EI]                                  CFA        985    A
CFA                                                                                CFA        986    A
CFA       FIS30    LX3    -ET.SUCCP                                                CFA        987    A
CFA                SX6    B4+X3                                                    CFA        988    A
CFA                ZR     X6,FIS31    IF SUCC[SI] = BN[IP]                         CFA        989    A
CFA                SB3    B3-B1       NS = NS - 1                                  CFA        990    A
CFA                SA3    A3+B1       SI = SI + 1                                  CFA        991    A
CFA                NZ     B3,FIS30    IF NS " 0                                    CFA        992    A
CFA                                                                                CFA        993    A
CFA                BX6    X5                                                       CFA        994    A
CFA                SA6    A5-B1       [IP-1] = [IP]                                CFA        995    A
CFA                SA7    A5          [IP] = LIW                                   CFA        996    A
CFA                BX5    X7                                                       CFA        997    A
CFA                SB6    B1          SWAP = 1                                     CFA        998    A
CFA                                                                                CFA        999    A
CFA       FIS31    BX7    X5          LIW = [IP]                                   CFA       1000    A
CFA                LX5    -IM.BNP                                                  CFA       1001    A
CFA                SB5    X5                                                       CFA       1002    A
CFA                NZ     B2,FIS29    IF N " 0                                     CFA       1003    A
CFA                ZR     B6,FIS32    IF SWAP = 0   */ NO INTERCHANGES             CFA       1004    A
CFA                SA5    A0          IP = IPF                                     CFA       1005    A
CFA                LX5    -IP.NODEP                                                CFA       1006    A
CFA                SB2    X5-2        N = NODE[IP] - 2                             CFA       1007    A
CFA                EQ     FIS28                                                    CFA       1008    A
CFA                                                                                CFA       1009    A
CFA       FIS32    SA5    A5+B1       IP = IP + 1                                  CFA       1010    A
CFA                NZ     X5,FIS27    IF [IP] " 0                                  CFA       1011    A
CFA       CBD      TITLE  CBD - COMPUTE BACK DOMINATORS                            CFA       1012    A
CFA      **        CBD - COMPUTE BACK DOMINATORS                                   CFA       1013    A
CFA      *                                                                         CFA       1014    A
CFA      *         RESCAN THE INTERVAL LIST AND COMPUTE THE BACK DOMINATORS AND    CFA       1015    A
CFA      *         INTERVAL ARTICULATION POINTS. THE FORMULAS USED ARE -           CFA       1016    A
CFA      *         BD(B) = & ( IP ! BD(IP) ) , IP  AN IMMEDIATE PRED OF B          CFA       1017    A
CFA      *         AP(INT) = & ( B ! BD(B) ) , B AN EXIT NODE                      CFA       1018    A
CFA      *         THE BIT VECTORS BVT(I) = I ! BD(I)                              CFA       1019    A
CFA                                                                                CFA       1020    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA      *         SETUP BLOCK TO INTERVAL TRANSITION TABLE                    CFA      1021     A
CFA                                                                            CFA      1022     A
CFA       CBD      MX6     0                                                   CFA      1023     A
CFA                TRACE   FIS.X,GST                                           CFA      1024     A
CFA                SA6     L.CFT        L.CFT = 0                              CFA      1025     A
CFA                SA2     N.BLK                                               CFA      1026     A
CFA                ALLOC   CFT,X2+B1    ALLOC( CFT , N.BLK + 1 )               CFA      1027     A
CFA                SA4     O.GST                                               CFA      1028     A
CFA                SA5     ILB                                                 CFA      1029     A
CFA                S"T"    X2           T = O.CFT                              CFA      1030     A
CFA                IX6     X4+X5                                               CFA      1031     A
CFA                SA5     X6           II = O.GST + ILB                       CFA      1032     A
CFA                SX0     B1                                                  CFA      1033     A
CFA                SB2     B0           MB = 0        */ MAX N.BLOCKS IN INTERVAL CFA      1034     A
CFA                LX0     I.WIP                                               CFA      1035     A
CFA                SX1     B1                                                  CFA      1036     A
CFA                LX1     I.NIP                                               CFA      1037     A
CFA                SB5     48                                                  CFA      1038     A
CFA                                                                            CFA      1039     A
CFA       CBD1     LX5     -IP.NODEP                                           CFA      1040     A
CFA                SB3     X5           NN = NODE[II]                          CFA      1041     A
CFA                SB4     B0           BITN = 0                               CFA      1042     A
CFA                MX2     0            WI = NI = 0                            CFA      1043     A
CFA                SA5     A5+B1        II = II + 1                            CFA      1044     A
CFA                GE      B2,B3,CBD2   MB = MAX( MB , NN )                    CFA      1045     A
CFA                SB2     B3+                                                 CFA      1046     A
CFA                                                                            CFA      1047     A
CFA       CBD2     PX6     B4,X2                                               CFA      1048     A
CFA                LX5     -IM.BNP                                             CFA      1049     A
CFA                SA6     "T"+X5       T(BN[II]) = TTW(BITN,WI,NI)            CFA      1050     A
CFA                SB4     B4+B1        BITN = BITN + 1                        CFA      1051     A
CFA                IX2     X2+X1        NI = NI + 1                            CFA      1052     A
CFA                LT      B4,B5,CBD3   IF BITN < 48                           CFA      1053     A
CFA                IX2     X2+X0        WI = WI + 1                            CFA      1054     A
CFA                SB4     B0           BITN = 0                               CFA      1055     A
CFA       CBD3     SB3     B3-B1        NN = NN - 1                            CFA      1056     A
CFA                SA5     A5+B1        II = II + 1                            CFA      1057     A
CFA                NZ      B3,CBD2      IF NN " 0                              CFA      1058     A
CFA                                                                            CFA      1059     A
CFA                NZ      X5,CBD1      IF [II] " 0   */ NOT END OF INTERVALS  CFA      1060     A
CFA                                                                            CFA      1061     A
CFA      *         ALLOCATE SPACE FOR A BIT VECTOR TABLE                       CFA      1062     A
CFA                                                                            CFA      1063     A
CFA                SX1     B2+47                                               CFA      1064     A
CFA                SX2     B5                                                  CFA      1065     A
CFA                IX7     X1/X2,B6     VM = (MB+47) / 48                      CFA      1066     A
CFA                SX2     B2+B1                                               CFA      1067     A
CFA                IX1     X2*X7                                               CFA      1068     A
CFA                ALLOC   RND,X1       ALLOC( RND , (MB+1)*VM )               CFA      1069     A
CFA                                                                            CFA      1070     A
CFA                SA1     O.GST                                               CFA      1071     A
CFA                SA3     ILB                                                 CFA      1072     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA                SA5    EII                                                    CFA        1073     A
CFA                SB3    X1                                                     CFA        1074     A
CFA                S"BV"  X2          BV = O.RND    */ BASE OF BIT VECTOR TABLE   CFA        1075     A
CFA                S"EI"  B3+X5       EIB = O.GST + EII                            CFA        1076     A
CFA                SA2    O.CFT                                                  CFA        1077     A
CFA                SA4    B3+X3       IP = O.GST + ILB                             CFA        1078     A
CFA                S"T"   X2          T = O.CFT                                    CFA        1079     A
CFA                                                                              CFA        1080     A
CFA      *         PROCESS NEXT INTERVAL                                          CFA        1081     A
CFA                                                                              CFA        1082     A
CFA       CBD4     LX4    -IP.NODEP                                              CFA        1083     A
CFA                S"IB"  A4+B1       IBA = IP + 1  */ INTERVAL BASE ADDR ( ( HNA ) CFA        1084     A
CFA                SB2    X4          NN = NODE[IP]                                CFA        1085     A
CFA                SX6    "IB"+B2     IP = IBA + NN                                CFA        1086     A
CFA                SA6    IP                                                     CFA        1087     A
CFA                NE     B2,B1,CBD6  IF NN " 1                                    CFA        1088     A
CFA                                                                              CFA        1089     A
CFA       CBD5     SA3    IP                                                     CFA        1090     A
CFA                SA4    X3+                                                    CFA        1091     A
CFA                NZ     X4,CBD4     IF [IP] " 0   */ IF NOT END OF LIST          CFA        1092     A
CFA                MX7    0                                                      CFA        1093     A
CFA                SA7    L.CFT       L.CFT = 0                                    CFA        1094     A
CFA                SA7    L.RND                                                  CFA        1095     A
CFA                TRACE  CBD.X,GST                                              CFA        1096     A
CFA                EQ     FIS                                                    CFA        1097     A
CFA                                                                              CFA        1098     A
CFA      *         PROCESS INTERVAL WITH @ 48 NODES                               CFA        1099     A
CFA                                                                              CFA        1100     A
CFA       CBD6     SB3    48                                                     CFA        1101     A
CFA                GT     B2,B3,CBD12 IF NN > 48                                 CFA        1102     A
CFA                                                                              CFA        1103     A
CFA                SA0    B2-B1       NN = NN - 1                                  CFA        1104     A
CFA                SA5    "IB"+B1     II = IBA + 1  */ INDEX TO INTERVAL           CFA        1105     A
CFA                SX0    B1                                                     CFA        1106     A
CFA                MX7    60          AN = -0                                    CFA        1107     A
CFA                SX6    B1                                                     CFA        1108     A
CFA                SA6    "BV"        BV(0) = 1     */ HEADER NODE                 CFA        1109     A
CFA                                                                              CFA        1110     A
CFA      *         COMPUTE BACK DOMINATORS OF NODES                               CFA        1111     A
CFA      *         BD = & BVT(IP)                                                 CFA        1112     A
CFA                                                                              CFA        1113     A
CFA       CBD7     LX5    -IM.BNP                                                CFA        1114     A
CFA                SA3    "EI"+X5     EI = EIB + BN[II]                            CFA        1115     A
CFA                LX3    -EI.PRIP                                               CFA        1116     A
CFA                UX6    B3,X3       N = NP[EI]                                   CFA        1117     A
CFA                SA4    "EI"+X3     PI = EIB + PRI[EI]                           CFA        1118     A
CFA                SB2    A5-"IB"     I = II - IBA                                 CFA        1119     A
CFA                MX1    60          BD = -0                                    CFA        1120     A
CFA                                                                              CFA        1121     A
CFA       CBD8     LX4    -ET.PREDP                                              CFA        1122     A
CFA                SA3    "T"+X4                                                 CFA        1123     A
CFA                LX3    -I.NIP      J = NI[T(PRED[PI])]                          CFA        1124     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA              SA2    "BV"+X3                                          CFA        1125    A
CFA              SB3    B3-B1       N = N - 1                            CFA        1126    A
CFA              BX1    X1*X2       BD = BD & BVT(J)                     CFA        1127    A
CFA              SA4    A4+B1       PI = PI + 1                          CFA        1128    A
CFA              NZ     B3,CBD8     IF N " 0                             CFA        1129    A
CFA                                                                     CFA        1130    A
CFA              LX2    B2,X0                                            CFA        1131    A
CFA              BX6    X1+X2       BVT(I) = BD ! SHIFT(1,I)             CFA        1132    A
CFA              LX5    IM.BNP+59-IM.ENP                                 CFA        1133    A
CFA              SA6    "BV"+B2                                          CFA        1134    A
CFA              PL     X5,CBD9     IF ^EN[II]                           CFA        1135    A
CFA              BX7    X6*X7       AN = AN & BVT(I)                     CFA        1136    A
CFA                                                                     CFA        1137    A
CFA      *       CALCULATE IMMEDIATE BACK DOMINATOR ORDINAL              CFA        1138    A
CFA                                                                     CFA        1139    A
CFA       CBD9   PX1    X1                                               CFA        1140    A
CFA              LX5    1+IM.ENP                                         CFA        1141    A
CFA              SB3    47                                               CFA        1142    A
CFA              NX2    B2,X1       FBD = 47-NORMC( PACK(BD) )           CFA        1143    A
CFA              SX4    B3-B2                                            CFA        1144    A
CFA              SA2    "IB"+X4                                          CFA        1145    A
CFA              LX4    IM.IBDP                                          CFA        1146    A
CFA              BX6    X4+X5       IBD[II] = FBD                        CFA        1147    A
CFA              SA6    A5                                               CFA        1148    A
CFA              LX0    IM.DOMP                                          CFA        1149    A
CFA              BX6    X0+X2       DOM[IBA+FBD] = 1                     CFA        1150    A
CFA              LX0    -IM.DOMP                                         CFA        1151    A
CFA              SA6    A2                                               CFA        1152    A
CFA                                                                     CFA        1153    A
CFA              SB2    A0-B1       NN = NN - 1                          CFA        1154    A
CFA              SA0    A0-B1                                            CFA        1155    A
CFA              SA5    A5+B1       II = II + 1                          CFA        1156    A
CFA              NZ     B2,CBD7     IF NN " 0                            CFA        1157    A
CFA                                                                     CFA        1158    A
CFA      *       PROPAGATE *AN* BIT VECTOR TO INTERVAL LIST              CFA        1159    A
CFA                                                                     CFA        1160    A
CFA              SA4    "IB"                                             CFA        1161    A
CFA              LX4    59-IM.ENP                                        CFA        1162    A
CFA              MI     X4,CBD5     IF EN[IBA]    */ HEADER IS AN EXIT NODE  CFA    1163    A
CFA              ZR     X7,CBD5     IF AN = 0                            CFA        1164    A
CFA                                                                     CFA        1165    A
CFA              SX1    B1                                               CFA        1166    A
CFA              LX1    IM.APP                                           CFA        1167    A
CFA                                                                     CFA        1168    A
CFA       CBD11  PX5    X7                                               CFA        1169    A
CFA              NX6    B2,X5                                            CFA        1170    A
CFA              SB2    B3-B2       J = 47 - NORMC(AN)                   CFA        1171    A
CFA              SA2    "IB"+B2                                          CFA        1172    A
CFA              LX4    B2,X0                                            CFA        1173    A
CFA              BX6    X1+X2       AP[IBA+J] = 1                        CFA        1174    A
CFA              BX7    -X4*X7      AN = AN & ^SHIFT(1,J)                CFA        1175    A
CFA              SA6    A2                                               CFA        1176    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA                NZ     X7,CBD11    IF AN " 0                                          CFA    1177    A
CFA                                                                                      CFA    1178    A
CFA                EQ     CBD5                                                            CFA    1179    A
CFA        CBD     SPACE  3,14                                                            CFA    1180    A
CFA      *         NODE[INTERVAL] > 48 , MULTI WORD BIT VECTOR CALCULATION                 CFA    1181    A
CFA                                                                                      CFA    1182    A
CFA        VL      MICRO  1,,/X1/     VECTOR LENGTH                                        CFA    1183    A
CFA        BI      MICRO  1,,/"BV"/   BVT(I) DURING MULTI WORD CALC                        CFA    1184    A
CFA                                                                                      CFA    1185    A
CFA        CBD12   SX6    B2-B1       NN = NN - 1                                          CFA    1186    A
CFA                SA6    NN                                                              CFA    1187    A
CFA                SX2    B3                                                              CFA    1188    A
CFA                SX1    X6+B3                                                           CFA    1189    A
CFA                I"VL"  X1/X2,B3    VL = (NN+48)/48  */ VECTOR LENGTH                   CFA    1190    A
CFA                                                                                      CFA    1191    A
CFA      *         INITIALIZE BIT VECTORS FOR ARTICULATION POINTS AND HEADER NODE          CFA    1192    A
CFA                                                                                      CFA    1193    A
CFA                MX6    60                                                              CFA    1194    A
CFA                SA6    "BV"        BVT(0) = -0                                         CFA    1195    A
CFA                SX7    B1                                                              CFA    1196    A
CFA                SA7    "BV"+"VL"   BVT(VL) = 1                                         CFA    1197    A
CFA                MX7    0                                                               CFA    1198    A
CFA                SB2    X1          FOR I = 1 TO VL - 1,                                CFA    1199    A
CFA                                                                                      CFA    1200    A
CFA      +         SA6    A6+B1       BVT(I) = 0                                          CFA    1201    A
CFA                SB2    B2-1                                                            CFA    1202    A
CFA                SA7    A7+B1       BVT(I+VL) = 0                                       CFA    1203    A
CFA                GT     B2,B1,*-1   END                                                 CFA    1204    A
CFA                                                                                      CFA    1205    A
CFA                SA5    "IB"                                                            CFA    1206    A
CFA                SA0    "BV"                                                            CFA    1207    A
CFA                S"BI"  "BV"+"VL"   BI = O.BVT + VL */ INDEX TO BIT VECTOR              CFA    1208    A
CFA                LX5    -IM.BNP                                                         CFA    1209    A
CFA                SA4    "T"+X5                                                          CFA    1210    A
CFA                SX6    "BI"                                                            CFA    1211    A
CFA                LX6    I.BVIP                                                          CFA    1212    A
CFA                BX7    X4+X6       BVI[T(BN[IBA])] = BI                                CFA    1213    A
CFA                SA7    A4                                                              CFA    1214    A
CFA                S"BI"  "BI"+"VL"   BI = BI + VL                                        CFA    1215    A
CFA                SA5    A5+B1       II = IBA + 1  */ INTERVAL INDEX                     CFA    1216    A
CFA                                                                                      CFA    1217    A
CFA      *         CALCULATE BACK DOMINATOR OF THIS NODE                                   CFA    1218    A
CFA                                                                                      CFA    1219    A
CFA        CBD14   LX5    -IM.BNP                                                         CFA    1220    A
CFA                SA3    "EI"+X5     EI = EIB + BN[II]                                   CFA    1221    A
CFA                LX3    -EI.PRIP                                                        CFA    1222    A
CFA                UX6    B3,X3       N = NP[EI]                                          CFA    1223    A
CFA                SA4    "EI"+X3     PI = EIB + PRI[EI]                                  CFA    1224    A
CFA                LX4    -ET.PREDP                                                       CFA    1225    A
CFA                SA3    "T"+X4      TW = T(PRED[PI])                                    CFA    1226    A
CFA                SB2    "VL"        I = VL                                              CFA    1227    A
CFA                LX3    -I.BVIP                                                         CFA    1228    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA              SA2    X3          BJ = BVT[TW]                                CFA    1229    A
CFA              BX6    X2                                                      CFA    1230    A
CFA              SA6    "BI"        BK = BI;  [BK] = [BJ]                       CFA    1231    A
CFA       CBD15  SA2    A2+B1       BJ = BJ + 1                                 CFA    1232    A
CFA              SB2    B2-B1       I = I - 1                                   CFA    1233    A
CFA              BX6    X2                                                      CFA    1234    A
CFA              SA6    A6+B1       BK = BK + 1;  [BK] = [BJ]                   CFA    1235    A
CFA              GT     B2,B1,CBD15 IF I > 1                                    CFA    1236    A
CFA              EQ     B3,B1,CBD17 IF N = 1                                    CFA    1237    A
CFA                                                                             CFA    1238    A
CFA       CBD16  SA4    A4+B1       PI = PI + 1                                 CFA    1239    A
CFA              LX4    -ET.PREDP                                               CFA    1240    A
CFA              SA3    "T"+X4                                                  CFA    1241    A
CFA              LX3    -I.BVIP                                                 CFA    1242    A
CFA              SB2    "VL"                                                    CFA    1243    A
CFA              SA2    "BI"                                                    CFA    1244    A
CFA              RJ     ABV         BD(I) = BD(I) & BVT(PRED)                   CFA    1245    A
CFA              SB3    B3-B1       N = N - 1                                   CFA    1246    A
CFA              GT     B3,B1,CBD16 IF N > 1                                    CFA    1247    A
CFA                                                                             CFA    1248    A
CFA      *       CALCULATE IMMEDIATE BACK DOMINATOR                             CFA    1249    A
CFA                                                                             CFA    1250    A
CFA       CBD17  SA4    "BI"+"VL"   BJ = BI + VL                                CFA    1251    A
CFA              SB3    47                                                      CFA    1252    A
CFA                                                                             CFA    1253    A
CFA       CBD18  SA4    A4-B1       BJ = BJ - 1                                 CFA    1254    A
CFA              ZR     X4,CBD18    IF [BJ] = 0                                 CFA    1255    A
CFA              PX6    X4                                                      CFA    1256    A
CFA              NX3    B2,X6                                                   CFA    1257    A
CFA              SX2    A4-"BI"                                                 CFA    1258    A
CFA              SB2    B3-B2                                                   CFA    1259    A
CFA              LX2    4                                                       CFA    1260    A
CFA              SX0    B1                                                      CFA    1261    A
CFA              LX3    B1,X2                                                   CFA    1262    A
CFA              IX6    X2+X3                                                   CFA    1263    A
CFA              LX0    IM.DOMP                                                 CFA    1264    A
CFA              SX2    B2+X6       FBD = 47 - NORMC([BJ]) + 48*( BJ - BI )     CFA    1265    A
CFA              SA3    "IB"+X2                                                 CFA    1266    A
CFA              BX6    X0+X3       DOM[IBA+FBD] = 1                            CFA    1267    A
CFA              SA6    A3                                                      CFA    1268    A
CFA                                                                             CFA    1269    A
CFA              SA4    "T"+X5      TW = T(BN[II])                              CFA    1270    A
CFA              LX5    IM.BNP                                                  CFA    1271    A
CFA              LX2    IM.IBDP                                                 CFA    1272    A
CFA              BX6    X2+X5       IBD[II] = FBD                               CFA    1273    A
CFA              SA6    A5                                                      CFA    1274    A
CFA              SX3    "BI"                                                    CFA    1275    A
CFA              LX3    I.BVIP                                                  CFA    1276    A
CFA              BX6    X3+X4       BVI[TW] = BI  */ INSTALL BIT VECTOR ADDRESS CFA    1277    A
CFA              SA6    A4                                                      CFA    1278    A
CFA              UX4    B3,X4                                                   CFA    1279    A
CFA              SX0    B1                                                      CFA    1280    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA                LX4    -I.WIP      W = WI[TW]                                    CFA    1281    A
CFA                SA2    "BI"+X4                                                   CFA    1282    A
CFA                LX0    B3,X0                                                     CFA    1283    A
CFA                BX7    X0+X2       BVT(BI+W) = BVT(BI+W) ! SHIFT(1,BITN[TW])     CFA    1284    A
CFA                SA7    A2                                                        CFA    1285    A
CFA                LX5    59-IM.ENP                                                 CFA    1286    A
CFA                PL     X5,CBD19    IF ^EN[II]    */ IF NOT AN EXIT NODE          CFA    1287    A
CFA                SB2    "VL"                                                      CFA    1288    A
CFA                SA2    A0                                                        CFA    1289    A
CFA                SX3    "BI"                                                      CFA    1290    A
CFA                RJ     ABV         AN = AN & BVT(BI)                             CFA    1291    A
CFA                                                                                CFA    1292    A
CFA       CBD19    SA2    NN                                                        CFA    1293    A
CFA                S"BI"  "BI"+"VL"   BI = BI + VL                                  CFA    1294    A
CFA                SA5    A5+B1       II = II + 1                                   CFA    1295    A
CFA                SX6    X2-1        NN = NN - 1                                   CFA    1296    A
CFA                SA6    A2+                                                       CFA    1297    A
CFA                NZ     X6,CBD14    IF NN " 0                                     CFA    1298    A
CFA                                                                                CFA    1299    A
CFA                S"BV"  A0                                                        CFA    1300    A
CFA                SA2    "IB"                                                      CFA    1301    A
CFA                LX2    59-IM.ENP                                                 CFA    1302    A
CFA                MI     X2,CBD5     IF EN[IBA]    */ IF HEADER IS AN EXIT NODE    CFA    1303    A
CFA                                                                                CFA    1304    A
CFA      *         PROPAGATE *AN* BIT VECTOR TO INTERVAL MEMBERS                     CFA    1305    A
CFA                                                                                CFA    1306    A
CFA                SB3    47                                                        CFA    1307    A
CFA                SA5    "BV"+"VL"   BI = O.BVT + VL                               CFA    1308    A
CFA                SX2    B3+B1                                                     CFA    1309    A
CFA                SX0    B1                                                        CFA    1310    A
CFA                IX2    X1*X2       WA = 48 * VL  */ WORD ADDEND                  CFA    1311    A
CFA                SX7    B1                                                        CFA    1312    A
CFA                LX7    IM.APP                                                    CFA    1313    A
CFA                                                                                CFA    1314    A
CFA       CBD20    SX2    X2-48                                                     CFA    1315    A
CFA                MI     X2,CBD5     IF WA < 0                                     CFA    1316    A
CFA                SA5    A5-B1       BI = BI - 1;  AN = [BI]                       CFA    1317    A
CFA                ZR     X5,CBD20    IF AN = 0                                     CFA    1318    A
CFA                                                                                CFA    1319    A
CFA       CBD21    PX6    X5                                                        CFA    1320    A
CFA                NX4    B2,X6                                                     CFA    1321    A
CFA                SB2    B3-B2       J = 47 - NORMC(AN)                            CFA    1322    A
CFA                LX3    B2,X0                                                     CFA    1323    A
CFA                SB2    B2+X2                                                     CFA    1324    A
CFA                BX5    -X3*X5      AN = ^SHIFT(1,J) & AN                         CFA    1325    A
CFA                SA3    "IB"+B2                                                   CFA    1326    A
CFA                BX6    X7+X3       AP[IBA+WA+J] = 1                              CFA    1327    A
CFA                SA6    A3                                                        CFA    1328    A
CFA                NZ     X5,CBD21    IF AN " 0                                     CFA    1329    A
CFA                EQ     CBD20                                                     CFA    1330    A
CFA                                                                                CFA    1331    A
CFA       FISA     BFMW   IM,(AP,HN)                                                CFA    1332    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA       FISB     VFD    2/3,40/FISB,18/377777B                                  CFA        1333     A
CFA       ABV      SPACE  2           ABV                                         CFA        1334     A
CFA      **        ABV - *AND* BIT VECTORS                                        CFA        1335     A
CFA      *                                                                        CFA        1336     A
CFA      *         ENTRY  (B2) = VL , BIT VECTOR LENGTH                           CFA        1337     A
CFA      *                (A2) = ADDRESS OF OPERAND/RESULT BIT VECTOR             CFA        1338     A
CFA      *                (X3) = ADDRESS OF SECOND OPERAND                        CFA        1339     A
CFA                                                                               CFA        1340     A
CFA       ABV1     BX6    X2*X3                                                   CFA        1341     A
CFA                SA3    A3+B1                                                   CFA        1342     A
CFA                SB2    B2-B1                                                   CFA        1343     A
CFA                SA6    A2                                                      CFA        1344     A
CFA                SA2    A2+1                                                    CFA        1345     A
CFA                GT     B2,ABV1                                                 CFA        1346     A
CFA       ABV      ROUTINE                                                        CFA        1347     A
CFA                SA3    X3                                                      CFA        1348     A
CFA                EQ     ABV1                                                    CFA        1349     A
CFA       FDG      TITLE  FDG - FORM DERIVED GRAPH                                CFA        1350     A
CFA      **        FDG - FORM DERIVED GRAPH OF A SET OF INTERVALS                 CFA        1351     A
CFA      *         THIS ROUTINE DEVIATES FROM THE NORMAL PROCEDURE THAT REDUCES   CFA        1352     A
CFA      *         EACH INTERVAL TO A POINT. INSTEAD IT REDUCES ONLY THE STRONGLY CFA        1353     A
CFA      *         CONNECTED REGIONS OF AN INTERVAL TO A POINT AND LEAVES THE     CFA        1354     A
CFA      *         REST OF THE BLOCKS ALONE.                                      CFA        1355     A
CFA                                                                               CFA        1356     A
CFA                DESCRIBE T.,60     T(OLD BN) = 30/NEW NODENUM,30/NEW BLOCK NUM CFA        1357     A
CFA       NN       DEFINE 30          NEW NODE NUMBER                             CFA        1358     A
CFA       NBN      DEFINE 30          NEW BLOCK NUMBER                            CFA        1359     A
CFA                                                                               CFA        1360     A
CFA       FDG      ROUTINE                                                        CFA        1361     A
CFA                SA5    N.BLK                                                   CFA        1362     A
CFA                ALLOC  OTI,X5+B1   ALLOC( OTI , N.BLK+1 )  */ TRANSITION TBL  CFA        1363     A
CFA                SA4    LC                                                      CFA        1364     A
CFA                SX0    X4+4                                                    CFA        1365     A
CFA                IX1    X0+X5                                                   CFA        1366          I
          -FCC0408
CFA                IX1    X0+X1                                                   FCC0408     107     A
CFA                ALLOC  CFT,X1      ALLOC( CFT , LC+5+N.BLK )                   CFA        1367     A
CFA                IX1    X0+X2                                                   CFA        1368     A
CFA                SB2    X1          O.T = O.CFT + LC+4  */ TEMP TABLE           CFA        1369     A
CFA                SA3    ILB                                                     CFA        1370     A
CFA                SA4    O.GST                                                   CFA        1371     A
CFA                SA0    X2          C0 = O.CFT                                  CFA        1372     A
CFA                SX6    X2+B1       O.CFT = O.CFT + 1                           CFA        1373     A
CFA                SA6    A2                                                      CFA        1374     A
CFA                SB4    X4                                                      CFA        1375     A
CFA                SA1    EII                                                     CFA        1376     A
CFA                SA5    B4+X3       IP = O.GST + ILB                            CFA        1377     A
CFA                SB4    B4+X1       EIB = O.GST + EII                           CFA        1378     A
CFA                MX6    0                                                       CFA        1379     A
CFA                SA6    B2          T(0) = 0                                    CFA        1380     A
CFA                SB6    1+IM.SCRP-IM.BIP                                       CFA        1381     A
CFA                                                                               CFA        1382     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA      *         CLEAR OUT TRANSITION TABLE IN CASE OF UNREACHABLE CODE          CFA      1383     A
CFA                                                                                CFA      1384     A
CFA               SA4    N.BLK       N = N.BLK;  I = 0                             CFA      1385     A
CFA               MX6    2                                                         CFA      1386     A
CFA       FDG0    SX4    X4-1        N = N - 1                                     CFA      1387     A
CFA               SA6    A6+1        I = I + 1;  T(I) = 1S58                       CFA      1388     A
CFA               NZ     X4,FDG0     IF N " 0                                      CFA      1389     A
CFA                                                                                CFA      1390     A
CFA      *         FORM THE TRANSITION TABLE, AND REASSIGN NODE NUMBERS TO THE     CFA      1391     A
CFA      *         BLOCKS THAT ARE CONSISTANT WITH THEIR PROGRAM ORDER.            CFA      1392     A
CFA      *         FIRST SCAN THE INTERVALS                                        CFA      1393     A
CFA                                                                                CFA      1394     A
CFA       FTT1    LX5    -IP.NODEP                                                 CFA      1395     A
CFA               SB3    X5          N = NODE[IP]                                  CFA      1396     A
CFA               LX5    IP.NODEP-IP.PBP                                           CFA      1397     A
CFA               SX7    X5          HB = PB[IP]   */ NEW *BN* OF *SCR*            CFA      1398     A
CFA               SA5    A5+B1       IP = IP + 1                                   CFA      1399     A
CFA               ZR     X7,FTT2     IF HB = 0     */ NO *SCR*                     CFA      1400     A
CFA                                                                                CFA      1401     A
CFA               LX5    -IM.BNP     HN = BN[IP]                                   CFA      1402     A
CFA               SA7    B2+X5       T(HN) = TTW(0,HB)                             CFA      1403     A
CFA               MX0    1                                                         CFA      1404     A
CFA               SX4    X5                                                        CFA      1405     A
CFA               BX7    X0+X7                                                     CFA      1406     A
CFA               LX4    T.NNP                                                     CFA      1407     A
CFA               BX7    X4+X7       TSW = TTW(1S29+HN,HB)                         CFA      1408     A
CFA               EQ     FTT4                                                      CFA      1409     A
CFA                                                                                CFA      1410     A
CFA       FTT2    LX5    59-IM.SCRP                                                CFA      1411     A
CFA               PL     X5,FTT3     IF ^SCR[IP]                                   CFA      1412     A
CFA               LX5    1+IM.SCRP-IM.BNP                                          CFA      1413     A
CFA               SA7    B2+X5       T(BN[IP]) = TSW                               CFA      1414     A
CFA               EQ     FTT4                                                      CFA      1415     A
CFA                                                                                CFA      1416     A
CFA       FTT3    LX3    B6,X5                                                     CFA      1417     A
CFA               LX5    1+IM.SCRP-IM.BNP                                          CFA      1418     A
CFA               SX6    X3                                                        CFA      1419     A
CFA               AX6    1                                                         CFA      1420     A
CFA               SA6    B2+X5       T(BN[IP]) = TTW(0,BI[IP]/2)                   CFA      1421     A
CFA                                                                                CFA      1422     A
CFA       FTT4    SB3    B3-B1       N = N - 1                                     CFA      1423     A
CFA               SA5    A5+B1       IP = IP + 1                                   CFA      1424     A
CFA               NZ     B3,FTT2     IF N " 0                                      CFA      1425     A
CFA               NZ     X5,FTT1     IF [IP] " 0                                   CFA      1426     A
CFA                                                                                CFA      1427     A
CFA      *         RESCAN THE TEMP TABLE AND ASSIGN NEW NODE NUMBERS TO THE NODES  CFA      1428     A
CFA      *         THAT ARE ^SCR OR HEADER.                                        CFA      1429     A
CFA                                                                                CFA      1430     A
CFA               SA2    N.BLK                                                     CFA      1431     A
CFA               SX0    B1                                                        CFA      1432     A
CFA               SB3    B1          I = 1                                         CFA      1433     A
CFA               SB5    X2                                                        CFA      1434     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA              LX0    T.NNP       J = 1                                  CFA    1435    A
CFA              BX1    X0                                                 CFA    1436    A
CFA                                                                        CFA    1437    A
CFA       FTT5   SA4    B2+B3                                              CFA    1438    A
CFA              MI     X4,FTT6     IF T(I) < 0   */ SCR AND NOT HEADER    CFA    1439    A
CFA              BX6    X0+X4                                              CFA    1440    A
CFA              SA6    A4+         NN[T(I)] = J                           CFA    1441    A
CFA              IX0    X0+X1       J = J + 1                              CFA    1442    A
CFA       FTT6   SB3    B3+B1       I = I + 1                              CFA    1443    A
CFA              LE     B3,B5,FTT5  IF I @ N.BLK                           CFA    1444    A
CFA                                                                        CFA    1445    A
CFA      *       NOW SET THE *NN* OF THE ENTRIES THAT ARE SCR & ^HEADER    CFA    1446    A
CFA                                                                        CFA    1447    A
CFA              MX6    1                                                  CFA    1448    A
CFA              SB3    B1          I = 1                                  CFA    1449    A
CFA              LX6    -1                                                 CFA    1450    A
CFA              SA6    B2          T(0) = 1S58                            CFA    1451    A
CFA                                                                        CFA    1452    A
CFA       FTT7   SA4    B2+B3                                              CFA    1453    A
CFA              PL     X4,FTT8     IF T(I) > 0                            CFA    1454    A
CFA              LX4    -T.NNP      OBN = NN[T(I)]                         CFA    1455    A
CFA              SA3    B2+X4                                              CFA    1456    A
CFA              BX6    X3          T(I) = T(OBN)                          CFA    1457    A
CFA              SA6    A4                                                 CFA    1458    A
CFA       FTT8   SB3    B3+B1       I = I + 1                              CFA    1459    A
CFA              LE     B3,B5,FTT7  IF I @ N.BLK                           CFA    1460    A
CFA                                                                        CFA    1461    A
CFA              MX6    0                                                  CFA    1462    A
CFA              SA6    B2          T(0) = 0                               CFA    1463    A
CFA                                                                        CFA    1464    A
CFA      *       SCAN TEMP TABLE AND SETUP *OTI* FOR USE BY *FIS*          CFA    1465    A
CFA                                                                        CFA    1466    A
CFA              SA1    O.OTI                                              CFA    1467    A
CFA              AX0    T.NNP                                              CFA    1468    A
CFA              BX6    X0                                                 CFA    1469    A
CFA              SA6    L.OTI       L.OTI = J                              CFA    1470    A
CFA              SB6    T.NNP                                              CFA    1471    A
CFA              SB3    X1                                                 CFA    1472    A
CFA              SA5    B2+X2       TI = O.T + NB                          CFA    1473    A
CFA              SX6    X6-1                                               CFA    1474    A
CFA              SA6    N.BLK                                              CFA    1475    A
CFA              MX7    0                                                  CFA    1476    A
CFA              SA7    B3          OTI(0) = 0                             CFA    1477    A
CFA                                                                        CFA    1478    A
CFA       FDG5   AX4    B6,X5                                              CFA    1479    A
CFA              SX6    X5                                                 CFA    1480    A
CFA              SA6    B3+X4       OTI(NN[TI]) = NBN[TI]                  CFA    1481    A
CFA              SA5    A5-B1       TI = TI - 1                            CFA    1482    A
CFA              NZ     X5,FDG5     IF [TI] " 0                            CFA    1483    A
CFA                                                                        CFA    1484    A
CFA      *       REDUCE THE GRAPH BY REDUCING THE *SCR*"S OF THE INTERVALS TO  CFA  1485  A
CFA      *       POINTS                                                    CFA    1486    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

```
CFA                                                                                CFA        1487     A
CFA                 SB7    B1          I = 1                                       CFA        1488     A
CFA                 SA7    A0          CI = C0                                     CFA        1489     A
CFA                 MX6    -ET.JPP                                                 CFA        1490     A
CFA                 SB6    T.NNP                                                   CFA        1491     A
CFA                 MX0    CF.FROML                                                CFA        1492     A
CFA                 LX0    CF.FROML+CF.FROMP                                       CFA        1493     A
CFA                                                                                CFA        1494     A
CFA        FDG6     SA4    B4+B7       EI = EIB + I                                CFA        1495     A
CFA                 SA2    B2+B7                                                   CFA        1496     A
CFA                 LX2    CF.FROMP-T.NNP                                          CFA        1497     A
CFA                 BX1    X0*X2       FE = NN[T(I)] */ FROM EDGE                  CFA        1498     A
CFA                 LX4    -EI.SUIP                                                CFA        1499     A
CFA                 UX7    B3,X4       N = NS[EI]                                  CFA        1500     A
CFA                 SA5    B4+X4       SI = EIB + SUI[EI]                          CFA        1501     A
CFA                 SB7    B7+1        I = I + 1                                   CFA        1502     A
CFA                 LE     B3,FDG9     IF N @ 0                                    CFA        1503     A
CFA                                                                                CFA        1504     A
CFA        FDG7     LX5    -ET.SUCCP                                               CFA        1505     A
CFA                 SA3    B2+X5                                                   CFA        1506     A
CFA                 LX3    CF.FROMP-T.NNP                                          CFA        1507     A
CFA                 BX2    X0*X3       TE = NN[T(SUCC[SI])]                        CFA        1508     A
CFA                 IX7    X1-X2                                                   CFA        1509     A
CFA                 LX2    CF.TOP-CF.FROMP                                         CFA        1510     A
CFA                 ZR     X7,FDG8     IF FE = TE                                  CFA        1511     A
CFA                                                                                CFA        1512     A
CFA                 BX5    X6*X5                                                   CFA        1513     A
CFA                 IX4    X1+X2                                                   CFA        1514     A
CFA                 BX7    X5+X4                                                   CFA        1515     A
CFA                 SA7    A7+B1       CI = CI + 1;  [CI] = CFTE(JP[SI], FE , TE )  CFA        1516     A
CFA                                                                                CFA        1517     A
CFA        FDG8     SB3    B3-B1       N = N - 1                                   CFA        1518     A
CFA                 SA5    A5+B1       SI = SI + 1                                 CFA        1519     A
CFA                 NZ     B3,FDG7     IF N " 0                                    CFA        1520     A
CFA                                                                                CFA        1521     A
CFA        FDG9     LE     B7,B5,FDG6  IF I @ NB                                   CFA        1522     A
CFA                                                                                CFA        1523     A
CFA                 SB3    A0                                                      CFA        1524     A
CFA                 SX6    A7-B3       L.CFT = CI - C0                             CFA        1525     A
CFA                 SA6    L.CFT                                                   CFA        1526     A
CFA                                                                                CFA        1527     A
CFA                 TRACE  FDG.X,OTI                                               CFA        1528     A
CFA                 EQ     FDG                                                     CFA        1529     A
CFA                                                                                CFA        1530     A
CFA                 QUAL                                                           CFA        1531     A
CFA                                                                                CFA        1532     A
CFA         DGS     EQU    /CODE/DGS                                               CFA        1533          I
           -CCGNC
CFA                                                                                CFA        1534          I
           -CCGNC
CFA         F.CFA   END                                                            CFA        1535          I
           -CCGNC

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN CFA

      CFA        DGS#     EQENT  /CODE/DGS                                          CCGNC        832     A
      CFA        CG$LABD  EQENT  /CODE/CG$LABD                                      CCG16        290     A
      CFA                 ENTRY  F$CFA                                              CCGNC        833     A
      CFA                                                                           CCGNC        834     A
      CFA        F$CFA    END                                                      CCGNC        835     A

           SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - CFA

                IDENTIFIER                TOTAL    ACTIVE

                CFA                        1535      1402
                FCC0392                      12        12
                FCC0408                      16        16
                FCC0448                       9         9
                FCC1191                      27        27
                FCC1319                       5         3
                CCG1                         22         4
                FCC1623X                      3         0
                CCG7                          1         0
                CCGNC                        27        14
                CCG16                        28        28
                CCG18                         5         4
                CCG26                        29        28
                CCG27                        36        36
                CCGA021                       4         4

                LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN UDT

      UDT       *DECK    UDT                                                      UDT            1     A
      UDT                IDENT  UDT                                               UDT            2     A
      UDT        UDT     TITLE  UDT - USE/DEF TABLE PROCESSING                    UDT            3     A
      UDT       *CALL    SSTCALL                                                  UDT            4     A
      UDT        B=UDT   RPVDEF                                                   UDT            5     A
      UDT                PASS2TM                                                  UDT            6           I
                -CCG1
      UDT                                                                         UDT            7           I
                -CCG1
      UDT       *IF      DEF,UDT,1                                                UDT            8           I
                -FCCT005
      UDT                TRACER UDI                                               UDT            9           I
                -FCCT005
      UDT        UDT     SPACE  2                                                 UDT           10     A
      UDT       **       UDT - USE/DEF TABLE INITIALIZATION                       UDT           11     A
      UDT       *                                                                 UDT           12     A
      UDT       *        MISCELLANEOUS ROUTINES FOR THE PREPROCESSING PHASE OF OPT=2 .  UDT      13     A
      UDT        UDT     SPACE  2                                                 UDT           14     A
      UDT                LIST   -R                                                UDT           15     A
      UDT                                                                         UDT           16     A
      UDT                TABLES BST,IOL,RND,RXI,TXT,UDI                           UDT           17           I
                -CCG1
      UDT                ECHO   1,Z=(BST,IOL,RND,RXI,TXT,UDI)                     CCG1         376           I
                -CCGNC

                0         1         2         3         4         5         6         7         8
                12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN UDT

```
UDT                EXT    O.Z,L.Z                                                  CCG1        377          I
          -CCGNC
UDT                TABLES (BST,IOL,LBT,RND,RXI,TXT,UDI)                            CCGNC       836     A
UDT                                                                                UDT          18     A
UDT                ECHO   4,A=(BIT,BVT),B=(BST,RXI)                                UDT          19     A
UDT                EXT    Z.B                                                      UDT          20          I
          -CCGNC
UDT       Z.A      EQU    Z.B                                                      UDT          21          I
          -CCGNC
UDT                EXT    Z$B                                                      CCGNC       837     A
UDT       Z$A      EQU    Z$B                                                      CCGNC       838     A
UDT       O.A      EQU    O.B                                                      UDT          22     A
UDT       L.A      EQU    L.B                                                      UDT          23     A
UDT                SPACE  2                                                        UDT          24     A
UDT                USE    //                                                       UDT          25          I
          -CCG18
UDT       O.UDB    BSS    100B        FWA OF USE/DEF HASH TABLE                    UDT          26          I
          -CCG18
UDT                USE    0                                                        UDT          27          I
          -CCG18
UDT                EXT    F.MEM                                                    CCG18       183     A
UDT       O.UDB    EQU    F.MEM                                                    CCG18       184     A
UDT                                                                                UDT          28     A
UDT                USE    /BVINFO/                                                 UDT          29     A
UDT       VL       BSS    1           BIT VECTOR LENGTH                            UDT          30     A
UDT       N.VAR    BSS    1           N.VARIABLES IN *UDT*                         UDT          31     A
UDT       BVL      BSS    1           VL - 1                                       UDT          32     A
UDT       RCV      BSS    1           =1 IF REFERENCED COMMON VARIABLES            UDT          33     A
UDT                USE    0                                                        UDT          34     A
UDT                                                                                UDT          35     A
UDT      *         BLOCK HEADER TRAILER WORD DEFINTIONS                            UDT          36     A
UDT                                                                                UDT          37     A
UDT       BH.LENP  EQU    0                                                        UDT          38     A
UDT       BH.BIP   EQU    36                                                       UDT          39     A
UDT       BT.LENP  EQU    0                                                        UDT          40     A
UDT                                                                                UDT          41     A
UDT      *CALL     CFTBLD                                                          UDT          42     A
UDT                                                                                UDT          43     A
UDT                LIST   X                                                        UDT          44     A
UDT                                                                                UDT          45     A
UDT      *CALL     UDTBLD                                                          UDT          46     A
UDT                LIST   R                                                        UDT          47     A
UDT       CG.INIT  TITLE  CG.INIT - INITIALIZE *CCG*                               CCG4         76          I
          -CCGNC
UDT      **        CG.INIT - INITIALIZE *CCG* PROCESSING                           CCG4         77          I
          -CCGNC
UDT       CG$INIT  TITLE  CG$INIT - INITIALIZE *CCG*                               CCGNC       839     A
UDT      **        CG$INIT - INITIALIZE *CCG* PROCESSING                           CCGNC       840     A
UDT      *                                                                         CCG4         78     A
UDT      *         ENTRY  HOST COMPILER STATIC TABLES INITIALIZED IN HIGH CORE     CCG4         79     A
UDT      *                (X1) = *LBT* ORDINAL OF BLOCK THAT *SLIST* IS IN.       CCG4         80     A
UDT                                                                                CCG4         81     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN UDT

```
UDT       CG.INIT  ENTRY.                                                    CCG4          82          I
         -CCGNC
UDT                SX6    =XCCG.X                                           CCG4          83          I
         -CCGNC
UDT                SA6    =XCG.MOX    CG.MOX = LOCF(CCG.X)                  CCG4          84          I
         -CCGNC
UDT       .T       IFNE   TEST,0                                            CCG4          85          I
         -CCGNC
UDT                SX7    =XPTC=                                            CCG4          86          I
         -CCGNC
UDT                SA7    =XCO.RDR    CO.RDR = LOCF(PTC=)                   CCG4          87          I
         -CCGNC
UDT       CG$INIT  ENTRY. **                                                CCGNC        841     A
UDT       .T       IFNE   TEST,0                                            CCGNC        842     A
UDT                SX7    =XCG$PTC                                          CCGNC        843     A
UDT                SA7    =XHO$RDR    HO$RDR = LOCF(CG$PTC)                 CCGNC        844     A
UDT       .T       ENDIF                                                    CCG4          88     A
UDT       .LCM     IFNE   .DAL,0                                            CCG21         74     A
UDT                SA2    =XHO$LCM                                          CCG21         75     A
UDT                PL     X2,INIT0    IF NOT LCM=G                          CCG21         76     A
UDT                SX6    B1                                                CCG21         77     A
UDT                LX6    D.L2++59-D.STP                                    CCG21         78          I
         -CCG35
UDT                LX6    D.L2P+59-D.STP                            001190  CCG35         70     A
UDT                SA6    =XLCM#G           */  SET FLAG FOR *SQZ*          CCG21         79     A
UDT       INIT0    BSS    0                                                 CCG21         80     A
UDT       .LCM     ENDIF                                                    CCG21         81     A
UDT                SA1    =XCP.AFLS                                         CCG7         135          I
         -CCG9
UDT                BX6    X1                                                CCG7         136          I
         -CCG9
UDT                SA2    =XCP.AFLS                                         CCG9         131          I
         -CCGNC
UDT                BX6    X2                                                CCG9         132          I
         -CCGNC
UDT                LX6    30                                                CCG7         137          I
         -CCGNC
UDT                SA6    =XCG.INFL   CG.INFL = SHIFT(CP.AFLS,30)           CCG7         138          I
         -CCGNC
UDT                                                                         CCG4          89     A
UDT      *         INITIALIZE TABLE MANAGER, ALLOCATE SPACE FOR *TXT*       CCG4          90     A
UDT                                                                         CCG4          91     A
UDT                SA2    =XHO$OPT                                          CCGA025       11     A
UDT                MI     X2,INIT0A   IF OPT = 2                            CCGA025       12     A
UDT                SX6    7                                                 CCGA025       13     A
UDT                SA6    =XATSNAT#   NUMBER OF OPT1 *ACTIVE* TABLES        CCGA025       14     A
UDT                SX6    0                                                 CCGA025       15     A
UDT                SA6    =XATSTOG#   DONT TOGGLE                           CCGA025       16     A
UDT                SX6    1000B                                             CCGA025       17     A
UDT                SA6    =XTHRESH#   TABLE MANAGER SLOP                    CCGA025       18     A
UDT                                                                         CCGA025       19     A
UDT       INIT0A   BSS    0                                                 CCGA025       20     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN UDT

```
UDT                IFNE   HC.2OC,0,2                                          CCG16        291     A
UDT                SX6    O.UDB                                               CCG16        292     A
UDT                SA6    =XLM#       LM = O.UDB                              CCG16        293     A
UDT                BX0    X1                                                  CCG4          92     A
UDT                CALL   CWS=                                                CCG4          93           I
          -CCGNC
UDT                CALL   CWS#                                                CCGNC        845     A
UDT                AX6    2                                                   CCG4          94     A
UDT                ALLOC  TXT,X6      ALLOC( TXT , CWS/4 )                    CCG4          95     A
UDT                SX6    4                                                   CCG4          96     A
UDT                SA6    A3          L.TXT = 4                               CCG4          97     A
UDT                SA1    =XHO$OPT                                            CCGA011      112     A
UDT                MI     X1,INIT0A   IF OPT 2                                CCGA011      113           I
          -CCGA025
UDT                SX6    7                                                   CCGA011      114           I
          -CCGA025
UDT                SA6    =XATSNAT#   ATSNAT = 7 FOR OPT = 1                   CCGA011      115           I
          -CCGA025
UDT                SX6    B0                                                  CCGA011      116           I
          -CCGA025
UDT                SA6    =XATSTOG#   AND DON'T TOGGLE                        CCGA011      117           I
          -CCGA025
UDT                SX6    1000B                                               CCGA011      118           I
          -CCGA025
UDT                SA6    =XTHRESH#   TABLE MANAGER SLOP                      CCGA011      119           I
          -CCGA025
UDT                                                                           CCG4          98     A
UDT       *        SETUP *LBL* AND INITIALIZE FLAGS FOR BLOCK PROCESSING      CCG4          99           I
          -CCG8
UDT                                                                           CCG4         100           I
          -CCG8
UDT                SA1    =XL.LBT                                             CCG4         101           I
          -CCG8
UDT                ALLOC  LBL,X1      ALLOC( LBL , L.LBT )                    CCG4         102           I
          -CCG8
UDT                SA4    =XO.LBT                                             CCG4         103           I
          -CCG8
UDT                MX7    -18                                                 CCG4         104           I
          -CCG8
UDT                SB3    X3                                                  CCG4         105           I
          -CCG8
UDT                SB2    B0          I = 0                                   CCG4         106           I
          -CCG8
UDT        INIT1   SA5    X4+B2                                               CCG4         107           I
          -CCG8
UDT                BX6    -X7*X5                                              CCG4         108           I
          -CCG8
UDT                SA6    X2+B2       LBL(I) = LOW18[LBT(I)]                  CCG4         109           I
          -CCG8
UDT                SB2    B2+B1       I = I + 1                               CCG4         110           I
          -CCG8
UDT                LT     B2,B3,INIT1 IF I  < L.LBT                           CCG4         111           I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN UDT

```
          -CCG8
UDT                                                                   CCG4        112          I
          -CCG8
UDT                 BX7    X0                                         CCG4        113          I
          -CCG8
UDT                 SB2    X2                                         CCG4        114          I
          -CCG8
UDT                 SA7    =XCG.LBO    CG.LBO = PARAM                 CCG4        115          I
          -CCG8
UDT                 SA5    B2+X7                                      CCG4        116          I
          -CCG8
UDT                 BX6    X5                                         CCG4        117          I
          -CCG8
UDT                 SA6    =XBLEN      BLEN = LBL(CG.LBO)             CCG4        118          I
          -CCG8
UDT       *         INITIALIZE FLAGS FOR BLOCK PROCESSING             CCG8        182     A
UDT                                                                   CCG8        183     A
UDT                 SA5    =XO.LBT                                    CCG8        184          I
          -CCG18
UDT                 SA5    =XF$LBT                                    CCG18       185          I
          -CCG31
UDT                 BX6    X0                                         CCG8        185          I
          -CCGA011
UDT        INIT0A   BX6    X0                                         CCGA011     120          I
          -CCGA025
UDT                 BX6    X0                                         CCGA025      21     A
UDT                 SB2    X5                                         CCG8        186          I
          -CCG31
UDT                 SA4    B2+X0                                      CCG8        187          I
          -CCG31
UDT                 SA4    =XF$LBT+X0  GET INITIAL LENGTH OF BLOCK    CCG31         2     A
UDT                 SX7    X4                                         CCG8        188     A
UDT                 SA6    =XCG.LBO    CG.LBO = PARAM                 CCG8        189          I
          -CCGNC
UDT                 SA6    =XCC$LBO    CC$LBO = PARAM                 CCGNC       846     A
UDT                 SA7    =XCG.BLEN                                  CCG8        190          I
          -CCGNC
UDT                 SA7    =XCC$BLEN                                  CCGNC       847     A
UDT                 AX4    18                                         CCG8        191     A
UDT                 SX6    X4                                         CCG8        192     A
UDT                 SA6    =XCG.PC                                    CCG8        193          I
          -CCGNC
UDT                 SA6    =XCC$PC                                    CCGNC       848     A
UDT                                                                   CCG4        119     A
UDT                 SA1    =XCO.OPT                                   CCG4        120          I
          -CCGNC
UDT                 SA1    =XHO$OPT                                   CCGNC       849          I
          -CCGA011
UDT                 PL     X1,INIT2    IF CO.OPT " 2                  CCG4        121          I
          -CCGNC
UDT                 PL     X1,INIT2    IF HO$OPT " 2                  CCGNC       850     A
UDT                                                                   CCG4        122     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN UDT

UDT       *         INITIALIZE TABLES FOR OPT=2                                    CCG4        123     A
UDT       *         ALLOCATE AND CLEAR A BASE TABLE FOR *UDI*                      CCG4        124     A
UDT                                                                                CCG4        125     A
UDT                 CALL   IMPA        INITIALIZE MASS I/O PROCESSING              CCG7        139         I
          -CCGNC
UDT                 CALL   IMPA#       INITIALIZE MASS I/O PROCESSING              CCGNC       851         I
          -CCG11
UDT                 SA1    =XCC$FT                                                 CCG11       140     A
UDT                 NZ     X1,INIT1    IF CC$FT " 0  */ NOT FIRST TIME             CCG11       141     A
UDT                 SX6    B1                                                      CCG11       142     A
UDT                 SA6    A1          CC$FT = 1                                   CCG11       143     A
UDT                 CLOSE  =XF.OPT,UNLOAD            ENSURE NO FILE FROM PREV STEP CCG11       144         I
          -CCG35
UDT                 CLOSE  =XF.OPT,UNLOAD,RCL  ENSURE NO FILE FROM PREV. STEP     001210  CCG35        71     A
UDT                                                                                CCGA012      10     A
UDT       *         REQUEST ECS FL FOR OPT=2.                                      CCGA012      11     A
UDT                                                                                CCGA012      12     A
UDT        #ECS     IFNE   CT.ECS,0                                                CCGA012      13     A
UDT                                                                                CCGA012      14     A
UDT        #OS      IFEQ   .OS,2       IF SCOPE 2                                  CCGA012      15     A
UDT                 MEMORY LCM,INITA,R                                             CCGA012      16     A
UDT                 SA1    INITA                                                   CCGA012      17     A
UDT                 BX6    X1                                                      CCGA012      18     A
UDT                 AX1    30                                                      CCGA012      19     A
UDT                 LX6    59-1        1S59 IF LCM REDUCE MODE                     CCGA012      20     A
UDT                 PL     X6,INIT05   IF FIXED ECS FL MODE                        CCGA012      21     A
UDT                 SX7    30000B                                                  CCGA012      22     A
UDT        #OS      ELSE                                                           CCGA012      23     A
UDT                 MEMORY LCM,INITB,R                                             CCGA012      24     A
UDT                 SA1    INITB                                                   CCGA012      25     A
UDT                 SA2    =XMAX.LCM                                               CCGA012      26     A
UDT                 AX1    30          MAX OBTAINABLE LCM                          CCGA012      27     A
UDT                 BX0    X1                                                      CCGA012      28     A
UDT                 MX6    X0-X2                                                   CCGA012      29     A
UDT        .FTN     IFEQ   HC.ID,2     IF FTN HOST
>>>>      F1037CC 11                                                               F1037CC      42     A
UDT                 SA1    =XCO.EC
>>>>      F1037CC 12                                                               F1037CC      43     A
UDT                 AX1    59
>>>>      F1037CC 13                                                               F1037CC      44     A
UDT                 BX6    X1*X6       ALLOW ECS USE IFF EC ON
>>>>      F1037CC 14                                                               F1037CC      45     A
UDT        .FTN     ENDIF
>>>>      F1037CC 15                                                               F1037CC      46     A
UDT                 SA6    A2          MAX.LCM = MIN(MAX OBT.,200000B)             CCGA012      30     A
UDT                 MEMORY LCM,INITA,R                                             CCGA012      31     A
UDT                 SA1    INITA                                                   CCGA012      32     A
UDT                 AX1    30          CURRENT LCM                                 CCGA012      33     A
UDT                 SA2    MAX.LCM                                                 CCGA012      34     A
UDT                 SX6    30000B                                                  CCGA012      35     A
UDT                 MX7    X2-X6       REQUEST MIN(MAX.LCM,30000B)                 CCGA012      36     A
UDT                                                                                CCGA012      37     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN UDT

```
UDT        #OS      ENDIF                                                    CCGA012      38    A
UDT                 IX3     X1-X7                                            CCGA012      39    A
UDT                 LX7     30                                               CCGA012      40    A
UDT                 PL      X3,INIT05   IF WE ALREADY HAVE 30K ECS           CCGA012      41    A
UDT                 SA7     INITA                                            CCGA012      42    A
UDT                 MEMORY LCM,INITA,R,,NOABT      REQUEST 30K               CCGA012      43    A
UDT                 SA1     INITA                                            CCGA012      44    A
UDT                 AX1     30                                               CCGA012      45    A
UDT                                                                          CCGA012      46    A
UDT        INIT05   BX7     X1                                               CCGA012      47    A
UDT                 SA7     =XCP.AFLL                                        CCGA012      48    A
UDT        #ECS     ENDIF                                                    CCGA012      49    A
UDT                                                                          CCG11       145    A
UDT        INIT1    CALL    IMPA#       INITIALIZE MASS I/O PROCESSING       CCG11       146    A
UDT                 SA5     =XLM        O.UDB = LM                           CCG4        126          I
           -CCGNC
UDT                 SA5     =XLM#                                            CCGNC       852    A
UDT                 SX1     =XCO.OBL+101B     S = 101B + CO.OBL              CCG4        127          I
           -CCGNC
UDT                 SX1     =XHO$OBL+101B     S = 101B + HO$OBL              CCGNC       853    A
UDT                 ALLOC   TXT,X1      ALLOC( TXT , S )                     CCG4        128    A
UDT                 IX6     X1+X5                                            CCG4        129    A
UDT                 SA6     A5          LM = LM + S                          CCG4        130    A
UDT                 SX7     4                                                CCG4        131    A
UDT                 SA6     A2          O.TXT = LM                           CCG4        132    A
UDT                 SA7     A3          L.TXT = 4                            CCG4        133    A
UDT                 SETZERO X5,100B     SETZERO( UDB , 100B )                CCG4        134    A
UDT                 ALLOC UDI,2         ALLOC( UDI , 2 )                     CCG4        135    A
UDT                                                                          CCG4        136    A
UDT                 ALLOC   BST,4       ALLOC( BST , 4 )                     CCG4        137    A
UDT                 SX1     B1                                               CCG4        138    A
UDT                 SX2     B1+B1                                            CCG4        139    A
UDT                 LX1     CF.FROMP                                         CCG4        140    A
UDT                 LX2     CF.TOP                                           CCG4        141    A
UDT                 BX1     X1+X2                                            CCG4        142    A
UDT                 ADDWRD CFT,X1       ADDWRD CFT , (1,2) )                 CCG4        143    A
UDT                 MX6     0                                                CCG27       106    A
UDT                 SA6     =XCC$OPF    CC$OPF = 0                           CCG27       107    A
UDT                 SA1     =XLCM.OA                                         CCG4        144          I
           -CCG7
UDT                 BX6     X1          O.LCM = LCM.OA                       CCG4        145          I
           -CCG7
UDT                 SA6     =XO.LCM                                          CCG4        146          I
           -CCG7
UDT                                                                          CCG4        147    A
UDT        *        SETUP *BOS*"S AT THE BEGINNING OF *TXT* AND *MOD*        CCG4        148    A
UDT                                                                          CCG4        149    A
UDT         SBOS    MACRO                                                    CCG4        150    A
UDT                 SA4     =XF.RDT+OC.BOS                                   CCG4        151          I
           -CCGNC
UDT                 SA4     F.RDT+OC.BOS                                     CCGNC       854    A
UDT                 R=      B2,OC.BOS                                        CCG4        152    A

           0         1         2         3         4         5         6         7         8
           12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN UDT

```
UDT                MX7    0                                                    CCG4        153     A
UDT                PX6    B2,X7                                                CCG4        154     A
UDT                SA6    X2                                                   CCG4        155     A
UDT                SA7    X2+B1                                                CCG4        156     A
UDT                BX6    X4                                                   CCG4        157     A
UDT                SA6    A7+B1                                                CCG4        158     A
UDT                SA7    A6+B1                                                CCG4        159     A
UDT                ENDM                                                        CCG4        160     A
UDT                                                                            CCG4        161     A
UDT       INIT2    SA2    =XO.TXT                                              CCG4        162     A
UDT                SBOS                                                        CCG4        163     A
UDT                ALLOC  MOD,4       ALLOC( MOD , 4 )                         CCG4        164     A
UDT                SBOS                                                        CCG4        165     A
UDT                EQ     CG.INIT                                              CCG4        166         I
          -CCGNC
UDT                EQ     CG$INIT                                              CCGNC       855     A
UDT                                                                            CCGA012      50     A
UDT       INITA    BSSZ   1                                                    CCGA012      51     A
UDT       INITB    VFD    30/-1,30/0                                           CCGA012      52     A
UDT       AUT      TITLE  AUT - ADJUST USE/DEF INDEX TABLE                     UDT          48     A
UDT      **        AUT - ADJUST USE/DEF INEX TABLE TO BRING TO WORKING FORM    UDT          49     A
UDT      *         SORT TABLE BY [IH,CR] TO CHAIN CLASS"S AND EXPAND TO 2 WDS/ENTRY  UDT      50     A
UDT                                                                            UDT          51     A
UDT      *         I. - TEMP SORT TABLE FORMAT                                 UDT          52     A
UDT                                                                            UDT          53     A
UDT                DESCRIBE I.,60                                              UDT          54     A
UDT                DEFINE 12                                                   UDT          55     A
UDT       IH       DEFINE 18                                                   UDT          56     A
UDT                DEFINE 11                                                   UDT          57     A
UDT       CR       DEFINE 1                                                    UDT          58     A
UDT       UI       DEFINE 18                                                   UDT          59     A
UDT                                                                            UDT          60     A
UDT       AUT      ENTRY.                                                      UDT          61         I
          -CCGNC
UDT       AUT      ENTRY. **,#                                                 CCGNC       856     A
UDT                SA5    L.UDI                                                UDT          62     A
UDT                SX7    X5-2                                                 UDT          63     A
UDT                NZ     X7,AUT0     IF L.UDI " 2  */ TABLE NON-EMPTY         UDT          64     A
UDT                ADDWRD UDI,1       ADDWRD( UDI , 1 )  */ DUMMY ENTRY        UDT          65     A
UDT       AUT0     BSS    0                                                    UDT          66     A
UDT       DBG      IF     DEF,/DEBUG/UDI                                       UDT          67         I
          -FCCT005
UDT       .T       IFNE   TEST,0                                               FCCT005       5     A
UDT                SA1    =XCO.SNAP                                            FCCT005       6         I
          -CCGNC
UDT                SA1    =XHO$SNAP                                            CCGNC       857     A
UDT                LX1    1R2                                                  FCCT005       7     A
UDT                PL     X1,AUTT     IF SNAP=2 NOT SELECTED                   FCCT005       8     A
UDT                DCALL  DMPUDI,([O.UDI],L.UDI)    PRINT SYMBOL DICTIONARY    UDT          68     A
UDT       DBG      ENDIF                                                       UDT          69         I
          -FCCT005
UDT       AUTT     BSS    0                                                    FCCT005       9     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN UDT

```
UDT        .T       ENDIF                                                    FCCT005      10     A
UDT                 SA5     L.UDI                                            UDT          70     A
UDT                 BX6     X5                                               UDT          71     A
UDT                 LX5     1                                                UDT          72     A
UDT                 SA6     N.VAR       N.VAR = L.UDI                        UDT          73     A
UDT                 ALLOC   UDI,X5+3    ALLOC( UDI , 2*N.VAR + 3 )           UDT          74     A
UDT                 SB2     B6-2        N = OLD(L.UDI-2)                     UDT          75     A
UDT                 BX6     X5                                               UDT          76     A
UDT                 SA6     A3          L.UDI = 2*N.VAR  */ EXTRA SPACE FOR SORT ONLY    UDT  77     A
UDT                 SB3     X2+B1                                            UDT          78     A
UDT                 SB7     B3+X5       T0 = O.UDI+1 + 2*N.VAR               UDT          79     A
UDT                 SX0     B1                                               UDT          80     A
UDT                 MX7     0                                                UDT          81     A
UDT                 SA5     X2+2        UI = O.UDI + 2                       UDT          82     A
UDT                 LX0     UA.CRP                                           UDT          83     A
UDT                 SB4     B1+B1                                            UDT          84     A
UDT                 SA7     B7-B1       TI = T0 - 1;  [TI] = 0               UDT          85     A
UDT                 SX4     B4+B4       I = 4                                UDT          86     A
UDT                 SX1     B2+                                              UDT          87     A
UDT                                                                          UDT          88     A
UDT       *         FORM SORT TABLE SO AS TO GROUP BY *IH"S*                 UDT          89     A
UDT                                                                          UDT          90     A
UDT        AUT1     BX3     X0*X5                                            UDT          91     A
UDT                 LX5     -UA.IHP                                          UDT          92     A
UDT                 LX3     I.CRP-UA.CRP                                     UDT          93     A
UDT                 SX2     X5                                               UDT          94     A
UDT                 IX6     X3+X4       SW = ITF(IH,CR,I)                    UDT          95     A
UDT                 LX2     I.IHP                                            UDT          96     A
UDT                 SA5     A5+B1       UI = UI + 1                          UDT          97     A
UDT                 BX7     X2+X6                                            UDT          98     A
UDT                 SB2     B2-B1       N = N - 1                            UDT          99     A
UDT                 SX4     X4+B4       I = I + 2                            UDT         100     A
UDT                 SA7     A7+B1       TI = TI + 1;  [TI] = SW              UDT         101     A
UDT                 NZ      B2,AUT1     IF N " 0                             UDT         102     A
UDT                                                                          UDT         103     A
UDT                 SA0     A7          TL = TI                              UDT         104     A
UDT                 CALL    SHL         SORT( UDI , [IH,CR,I] )              UDT         105         I
          -CCG1
UDT                 CALL    SHL=        SORT( UDI , [IH,CR,I] )              CCG1        378         I
          -CCGNC
UDT                 CALL    SHL#        SORT( UDI , [IH,CR,I] )              CCGNC       858     A
UDT        AUT      SPACE   3,14                                             UDT         106     A
UDT       *         NOW SETUP FINAL FORM OF USE/DEF INDEX TABLE WHERE ALL    UDT         107     A
UDT       *         CLASS MEMBERS POINT TO THEIR BASE MEMBER AND THE MEMBERS OF  UDT     108     A
UDT       *         THE CLASS ARE CHAINED FROM THE BASE MEMBER TO THE CLASS. UDT         109     A
UDT                                                                          UDT         110     A
UDT                 SA2     O.UDI                                            UDT         111     A
UDT                 SA3     O.SYM                                            UDT         112         I
          -CCG1
UDT                 SA3     =XO.SYM                                          CCG1        379         I
          -CCGNC
UDT                 SA3     O.SYM                                            CCGNC       859     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN UDT

```
UDT                 SX0    B1                                                  UDT          113     A
UDT                 SA1    AUTA                                                UDT          114     A
UDT                 LX0    I.CRP                                               UDT          115     A
UDT                 SA5    A0          TI = TL                                 UDT          116     A
UDT                 SB2    X2                                                  UDT          117     A
UDT                 SA0    X3                                                  UDT          118          I
          -CCG1
UDT                 SB4    X3+1                                                CCG1         380     A
UDT                 SB6    I.IHP-1                                             UDT          119     A
UDT                 SX3    B1                                                  FCC0807       17     A
UDT                 LX3    P.LOCF                                              FCC0807       18          I
          -CCG1
UDT                 LX3    WB.LOCFP                                            CCG1         381     A
UDT                                                                            UDT          120     A
UDT       AUT2      ZR     X5,AUT4     IF [TI] = 0   */ END OF TABLE           UDT          121     A
UDT                 BX7    X0*X5       C = CR[TI]                              UDT          122     A
UDT                 SX6    X5          I = UI[TI]                              UDT          123     A
UDT                 AX5    B6,X5                                               UDT          124     A
UDT                 SB3    X5          IHB = IH[TI]                            UDT          125          I
          -CCG1
UDT                 SB3    B4+X5       IHB = IH[TI]                            CCG1         382     A
UDT                 LX7    UD.CRP-I.CRP                                        UDT          126     A
UDT                 SA2    A0-B3       WORDA = [O.SYM-2*IHB]                   UDT          127          I
          -CCG1
UDT                 BX4    X1*X2                                               UDT          128          I
          -CCG1
UDT                 SA2    A2-B1       WORDB = WORDA - 1                       FCC0807       19          I
          -CCG1
UDT                 BX5    X3*X2                                               FCC0807       20          I
          -CCG1
UDT                 LX5    P.COM-P.LOCF                                        FCC0807       21          I
          -CCG1
UDT                 BX4    X4+X5       COM = COM ! LOCF                        FCC0807       22          I
          -CCG1
UDT                 LX4    UD.COMP-P.COM                                       UDT          129          I
          -CCG1
UDT                 AX5    1                                                   CCG1         383     A
UDT                 SA2    B3+X5       WORDB = SYM(3*IHB+1)                    CCG1         384     A
UDT                 BX5    X3*X2                                               CCG1         385     A
UDT                 LX5    WB.COMP-WB.LOCFP                                    CCG1         386     A
UDT                 BX4    X1*X2                                               CCG1         387     A
UDT                 BX4    X4+X5       COM = COM ! LOCF                        CCG1         388     A
UDT                 LX4    UD.COMP-WB.COMP                                     CCG1         389     A
UDT                 SA5    A5-B1       TI = TI - 1                             UDT          130     A
UDT                 LX2    X7                                                  UDT          131     A
UDT                 BX7    X4+X7                                               UDT          132     A
UDT                 SA7    B2+X6       UDI(I) = UDIT(COM,C,0,0,0);  UL = UDI(I) UDT         133     A
UDT                 ZR     X2,AUT2     IF C = 0      */ NOT A CLASS REP        UDT          134     A
UDT                                                                            UDT          135     A
UDT       *         CHAIN CLASS MEMBERS FROM BASE, AND POINT THEM TO THE BASE MEMBER UDT    136     A
UDT                                                                            UDT          137     A
UDT                 LX2    UD.CMP-UD.CRP                                       UDT          138     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN UDT

```
UDT                BX4    X2+X4                                                    UDT          139     A
UDT                LX6    UD.BMIP                                                  UDT          140     A
UDT                IX6    X4+X6                                                    UDT          141     A
UDT                                                                                UDT          142     A
UDT       AUT3     AX4    B6,X5                                                    UDT          143     A
UDT                SB7    X4                                                       UDT          144         I
          -CCG20
UDT                SB7    X4+B4                                                    CCG20         33     A
UDT                NE     B7,B3,AUT2  IF IH[TI] " IHB                              UDT          145     A
UDT                SX2    X5          J = UI[TI]                                   UDT          146     A
UDT                SA5    A5-B1       TI = TI - 1                                  UDT          147     A
UDT                LX2    UD.ECLP                                                  UDT          148     A
UDT                BX7    X2+X7       ECL[UL] = J   */ LINK LAST TO THIS           UDT          149     A
UDT                SA7    A7                                                       UDT          150     A
UDT                LX2    -UD.ECLP                                                 UDT          151     A
UDT                BX7    X6                                                       UDT          152     A
UDT                SA7    B2+X2       UDI(J) = UDIT(COM,0,I,0);  UL = UDI(J)       UDT          153     A
UDT                EQ     AUT3                                                     UDT          154     A
UDT                                                                                UDT          155     A
UDT       AUTA     BFMW   UD,(FP,COM),P.COM-UD.COMP                                UDT          156         I
          -CCG1
UDT       AUTA     BFMW   WB,(FP,COM)                                              CCG1         390     A
UDT       AUT      SPACE  3,14                                                     UDT          157     A
UDT      *         FORM "SPOIL" BIT VECTOR FOR COMMON VARIABLES                    UDT          158     A
UDT                                                                                UDT          159     A
UDT       AUT4     SA5    N.VAR                                                    UDT          160     A
UDT                SX2    60                                                       UDT          161     A
UDT                SX1    X5+57                                                    UDT          162     A
UDT                IX7    X1/X2       L = (N.VAR+57) / 60                          UDT          163     A
UDT                LX2    B1,X7                                                    UDT          164     A
UDT                SA7    VL          VL = L                                       UDT          165     A
UDT                SX6    X7-1                                                     UDT          166     A
UDT                SA6    BVL         BVL = L - 1                                  UDT          167     A
UDT                LX7    2                                                        UDT          168     A
UDT                IX1    X2+X7                                                    UDT          169     A
UDT                ALLOC  BVT,X1      ALLOC( BVT , 6*L )                           UDT          170     A
UDT                                                                                UDT          171     A
UDT                SA4    O.UDI                                                    UDT          172     A
UDT                SB2    X5-2        N = N.VAR - 2                                UDT          173     A
UDT                SB4    -B1                                                      UDT          174     A
UDT                SB6    B0          BITN = 0                                     UDT          175     A
UDT                SB7    60                                                       UDT          176     A
UDT                MX7    0           WI = 0                                       UDT          177     A
UDT                PX7    B4,X7                                                    UDT          178     A
UDT                MX6    -UD.ISTP                                                 UDT          179     A
UDT                SA6    X4          UDI(0) = (DEFR,DEF,^RD,IST)                  UDT          180     A
UDT                SA7    X4+B1       UDI(1) = PACK(-1,0) */ NO REP IN BIT VECT    UDT          181     A
UDT                SA7    X4+3        UDI(3) = PACK(-1,0) */ FOR FIRST 2 ENTRIES   UDT          182     A
UDT                MX6    0           SV = 0                                       UDT          183     A
UDT                SA6    A7-B1       UDI(2) = 0                                   UDT          184     A
UDT                SX0    1                                                        UDT          185     A
UDT                SA5    A7+B1       UI = UDI(4)                                  UDT          186     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN UDT

```
UDT                MX1    0           SVS = 0       */ SPOIL VECTOR SUM                  UDT        187    A
UDT                LX0    UD.COMP                                                        UDT        188    A
UDT                SB3    UD.COMP     J = 0                                              UDT        189    A
UDT                                                                                      UDT        190    A
UDT        AUT5    PX7    B6,X7                                                          UDT        191    A
UDT                BX4    X0*X5                                                          UDT        192    A
UDT                SA7    A5+B1       US = UI + 1;  [US] = PACK(BITN,WI)                 UDT        193    A
UDT                SB2    B2-B1       N = N - 1                                          UDT        194    A
UDT                AX3    B3,X4       SV = SV ! SHIFT(1,COM[UI])                         UDT        195    A
UDT                SB3    B3-B1       J = J - 1                                          UDT        196    A
UDT                BX6    X3+X6                                                          UDT        197    A
UDT                SA5    A7+B1       UI = US + 1                                        UDT        198    A
UDT                SB6    B6+B1       BITN = BITN + 1                                    UDT        199    A
UDT                ZR     B2,AUT6     IF N = 0                                           UDT        200    A
UDT                LT     B6,B7,AUT5  IF BITN < 60                                       UDT        201    A
UDT                SB3    UD.COMP                                                        UDT        202    A
UDT                                                                                      UDT        203    A
UDT        AUT6    SA6    X2          [BI] = SV                                          UDT        204    A
UDT                BX1    X1+X6       SVS = SVS ! SV                                     UDT        205    A
UDT                SX2    X2+B1       BI = BI + 1                                        UDT        206    A
UDT                SB6    B0          BITN = 0                                           UDT        207    A
UDT                MX6    0           SV = 0                                             UDT        208    A
UDT                SX7    X7+B1       WI = WI + 1                                        UDT        209    A
UDT                NZ     B2,AUT5     IF N " 0                                           UDT        210    A
UDT                                                                                      UDT        211    A
UDT                CX7    X1                                                             UDT        212    A
UDT                SA7    RCV         RCV = COUNT(SVS)  */ REFED COMMON VARS FLAG        UDT        213    A
UDT        AUT     SPACE  1,10                                                           UDT        214    A
UDT      *         FORM DEAD DEFINITION CANIDATE COMPLEMENT VECTOR                        UDT        215    A
UDT      *         ^DDC = CR ! CM .                                                       UDT        216    A
UDT                                                                                      UDT        217    A
UDT                MX6    0           VW = 0                                             UDT        218         I
         -CCG1
UDT                SA5    BVL                                                            UDT        219    A
UDT                SA1    AUTB                                                           UDT        220    A
UDT                SB6    X5+B1       INDX = BVL + 1                                     UDT        221    A
UDT                SA0    X5+B1       L = BVL + 1                                        UDT        222    A
UDT                RJ     FSV         FORM *DDC* VECTOR                                  UDT        223    A
UDT                                                                                      UDT        224    A
UDT      *         FORM THE PROGRAM ENTRY/EXIT NODE BITS VECTORS FOR USE IN THE           UDT        225    A
UDT      *         LIVE EXIT BIT VECTOR COMPUTATION.  LET S = THE SET OF COMMON           UDT        226    A
UDT      *         VARS, F.P.S AND THE FUNCTION VALUE ( VALUE. ). I.E. THE SET            UDT        227         I
         -CCG1
UDT      *         VARIABLES ( LOCF VARS INCLUDED ) AND F.P.S .   I.E. THE SET            CCG1       391    A
UDT      *         OF VARIABLES WHICH ARE LIVE ON EXIT FROM THE SUBPROGRAM.               UDT        228    A
UDT      *         THEN DEF(0) = 0, AND UBD(0) = USE(0) = LX(0) = S .                     UDT        229    A
UDT                                                                                      UDT        230    A
UDT                SA2    O.BIT                                                          UDT        231    A
UDT                SX6    B6                                                             UDT        232    A
UDT                LX6    BI.BVIP                                                        UDT        233    A
UDT                SA6    X2          BVI[BIT(0)] = 2*L                                  UDT        234    A
UDT                SA6    X2+2        BVI[BIT(2)] = 2*L                                  UDT        235    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN UDT

```
UDT                MX7    0                                                    UDT          236     A
UDT                SB7    B7+A0                                                UDT          237     A
UDT      +         SA7    B5+B6       FOR I = 0 TO BVL; DEF(0) = 0             UDT          238     A
UDT                SB6    B6+B1                                                UDT          239     A
UDT                LT     B6,B7,*                                              UDT          240     A
UDT                SA2    =XFUNTYPE                                            UDT          241         I
         -CCG1
UDT                SB6    B6+A0       INDX = 4*L                               UDT          242         I
         -CCG1
UDT                MX6    0           VW = 0                                   UDT          243         I
         -CCG1
UDT                ZR     X2,AUT9     IF FUNTYPE = 0                           UDT          244         I
         -CCG1
UDT                SX3    X2-1        F = FUNTYPE                              UDT          245         I
         -CCG1
UDT                BX6    X2+X3       VW = F + (F-1)                           UDT          246         I
         -CCG1
UDT       AUT9     SA1    AUTC                                                 UDT          247         I
         -CCG1
UDT                SA1    AUTC                                                 CCG1         392     A
UDT                RJ     FSV         FORM EXIT BLOCK *USE* VECTOR             UDT          248     A
UDT                                                                            UDT          249     A
UDT      *         SET LX = USE AND UBD = USE                                  UDT          250     A
UDT                                                                            UDT          251     A
UDT                SB2    A0                                                   UDT          252     A
UDT                SB6    B6-B2       I = 4*L;  UL = 5*L                       UDT          253     A
UDT       AUT10    SA5    B5+B6       UVA = O.BVI + I;  VW = [UVA]             UDT          254     A
UDT                SB6    B6+B1       I = I + 1                                UDT          255     A
UDT                LX7    X5                                                   UDT          256     A
UDT                SA7    A5-B2       UFA = UVA - L;  [UFA] = VW               UDT          257         I
         -CCGA026
UDT                SA7    A5+B2       LXA = UVA + L;  [LXA] = VW               UDT          258         I
         -CCGA026
UDT                SA7    A5+B2       UFA = UVA + L       [UFA] = VW           CCGA026        6     A
UDT                SA7    A7+B2       LXA = UFA + L       [LXA] = VW           CCGA026        7     A
UDT                LT     B6,B7,AUT10 IF I < UL                                UDT          259     A
UDT                                                                            UDT          260     A
UDT                SA2    =XN.FP                                               UDT          261         I
         -CCG1
UDT                ZR     X2,AUT      IF N.FP = 0   */ NO F.P. S               UDT          262         I
         -CCG1
UDT                SA2    =XN.FPS                                              CCG1         393         I
         -CCGNC
UDT                SA2    =XN$FPS                                              CCGNC        860     A
UDT                ZR     X2,AUT      IF N.FPS = 0   */ NO F.P. S              CCG1         394         I
         -CCGNC
UDT                ZR     X2,AUT      IF N$FPS = 0   */ NO F.P. S              CCGNC        861     A
UDT                SA1    AUTC                                                 UDT          263     A
UDT                MX6    0           VW = 0                                   UDT          264         I
         -CCG1
UDT                SX7    B1                                                   UDT          265     A
UDT                SB6    B0          INDX = 0                                 UDT          266     A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN UDT

```
UDT                SA7    RCV         RCV = 1                                          UDT        267    A
UDT                RJ     FSV         INCLUDE F.P.S IN COMMON VAR SPOIL VECTOR         UDT        268    A
UDT                                                                                    UDT        269    A
UDT                EQ     AUT                                                          UDT        270    A
UDT                                                                                    UDT        271    A
UDT       AUTB     BFMW   UD,(CR,CM)                                                   UDT        272    A
UDT       AUTC     BFMW   UD,(FP,COM)                                                  UDT        273    A
UDT       FSV      SPACE  3,24        FSV                                              UDT        274    A
UDT      **        FSV - FORM SPECIAL BIT VECTOR FROM *UDI*                            UDT        275    A
UDT      *                                                                             UDT        276    A
UDT      *         ENTRY  (X1) = FM , MASK FOR BITS IN SET.                            UDT        277    A
UDT      *                (X6) = VW , FIRST WORD OF VECTOR ( PARTIAL )                UDT        278         I
         -CCG1
UDT      *                (A0) = L = BVL+1                                            UDT        279    A
UDT      *                (B6) = *BVT* STORE INDEX                                    UDT        280    A
UDT      *                                                                             UDT        281    A
UDT      *         EXIT   (B5) = [O.BVT]                                              UDT        282    A
UDT      *                (B6) = INDX + L                                             UDT        283    A
UDT                                                                                    UDT        284    A
UDT       FSV      ROUTINE                                                             UDT        285    A
UDT                SA2    N.VAR                                                       UDT        286    A
UDT                SA3    O.BVT                                                       UDT        287    A
UDT                SA4    O.UDI                                                       UDT        288    A
UDT                SB4    X2-2        N = N.VAR - 2                                   UDT        289    A
UDT                SB5    X3                                                          UDT        290    A
UDT                SB7    B6+A0       UL = INDX + L                                   UDT        291    A
UDT                SA5    X4+4        UI = [O.UDI] + 4                                UDT        292    A
UDT                SB2    B0          BITN = 0                                        UDT        293    A
UDT                SB3    60                                                          UDT        294    A
UDT                SX0    B1                                                          UDT        295    A
UDT                SX6    0           VW = 0                                          CCG1       395    A
UDT                                                                                    UDT        296    A
UDT       FSV1     BX7    X1*X5       B = FM[UI]                                      UDT        297    A
UDT                SA5    A5+2        UI = UI + 2                                     UDT        298    A
UDT                SB4    B4-B1       N = N - 1                                       UDT        299    A
UDT                ZR     X7,FSV2     IF B = 0      */ VAR NOT IN SET                 UDT        300    A
UDT                LX4    B2,X0                                                       UDT        301    A
UDT                BX6    X4+X6       VW = VW ! SHIFT( 1 , BITN )                     UDT        302    A
UDT       FSV2     SB2    B2+1        BITN = BITN + 1                                 UDT        303    A
UDT                ZR     B4,FSV3     IF N = 0                                        UDT        304    A
UDT                LT     B2,B3,FSV1  IF BITN < 60                                    UDT        305    A
UDT                SB2    0           BITN = 0                                        UDT        306    A
UDT       FSV3     SA6    B5+B6       BVT(INDX) = VW                                  UDT        307    A
UDT                SB6    B6+1        INDX = INDX + 1                                 UDT        308    A
UDT                MX6    0           VW = 0                                          UDT        309    A
UDT                LT     B6,B7,FSV1  IF INDX < UL                                    UDT        310    A
UDT                                                                                    UDT        311    A
UDT                EQ     FSV                                                         UDT        312    A
UDT       CMR      TITLE  CMR - CHAIN MEMORY REFERENCES TO *UDI*                      UDT        313    A
UDT      **        CMR - CHAIN MEMORY REFS IN *TXT* TO *UDI* BY THE *IN* FIELD        UDT        314    A
UDT      *         SETUP LINKS FROM INSTRUCTIONS TO *UDI* FOR LATER USE/DEF PROCESSING UDT        315    A
UDT      *                                                                             UDT        316    A

         0         1         2         3         4         5         6         7         8
         12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN UDT

```
UDT       *         AP AND IO LIST LD/ST"S ARE IGNORED                                UDT        317    A
UDT       *         F.P. ADDRESS LOADS ARE CHAINED TO ENTRY 1                         UDT        318    A
UDT                                                                                   UDT        319    A
UDT        CMR0     SX6    B6+         L.UDI  = UI                                    UDT        320    A
UDT                 SA6    L.UDI                                                      UDT        321    A
UDT                                                                                   UDT        322    A
UDT        CMR      ENTRY.                                                            UDT        323         I
          -CCGNC
UDT        CMR      ENTRY. **,#                                                       CCGNC      862    A
UDT                 SA1    L.TXT                                                      UDT        324    A
UDT                 AX1    2                                                          UDT        325    A
UDT                 ALLOC  UDI,X1      ALLOC( UDI , L.TXT/4 );  UI = OLD( L.UDI )     UDT        326    A
UDT                 SA5    O.TXT                                                      UDT        327    A
UDT                 SB7    X2          UDI = O.UDI                                    UDT        328    A
UDT                 MX0    -D.LDSTL                                                   UDT        329    A
UDT                 SB2    B1+B1                                                      UDT        330    A
UDT                 LX0    D.LDSTP                                                    UDT        331    A
UDT                 SA4    X5+B2       DI = O.TXT + 2                                 UDT        332    A
UDT                 MX1    -IH.CAIHL                                                  UDT        333    A
UDT                 LX1    IH.CAIHP                                                   UDT        334    A
UDT                 SA2    O.SYM                                                      FCC1088     27    A
UDT                 SA0    X2-1                     (A0) = SYM-1                      FCC1088     28         I
          -CCG1
UDT                 SB4    X2+1                     (B4) = SYM + 1                    CCG1       396    A
UDT                                                                                   UDT        335    A
UDT       *         SCAN *TXT* FOR MEMORY REFS AND CHAIN THEM TO *UDI*                UDT        336    A
UDT                                                                                   UDT        337    A
UDT        CMR1     SA5    A4+B2       R1 = DI + 2                                    UDT        338    A
UDT                 SA4    A5+B2       DI = R1 + 2                                    UDT        339    A
UDT                 UX7    B3,X5                                                      UDT        340    A
UDT                 BX6    -X0*X4                                                     UDT        341    A
UDT                 ZR     B3,CMR0     IF OC[R1] = 0 */ END OF BLOCK                  UDT        342    A
UDT                 ZR     X6,CMR1     IF LDST[DI] = 0  */ NOT A MEM REF              UDT        343    A
UDT                                                                                   UDT        344    A
UDT                 SA3    A5+B1       R2 = R1 + 1                                    UDT        345    A
UDT                 BX7    -X1*X3      IHF = CAIH[R2]                                 UDT        346    A
UDT                 MX2    -6                                                         UDT        347    A
UDT                 LX3    -IH.RFP                                                    UDT        348    A
UDT                 LX7    -IH.IHP                                                    UDT        349    A
UDT                 BX4    -X2*X7      HI = IHF & 77B  */ HASH INDEX                  UDT        350    A
UDT                 SB3    X3          C0 = RF[R2]                                    UDT        351    A
UDT                 NZ     B3,CMR6     IF RF[R2] " 0                                  UDT        352         I
          -CCG28
UDT                                                                                   UDT        353         I
          -CCG28
UDT                 SX6    X7-I.GL                                                    UDT        354         I
          -CCG1
UDT                 SA2    =XCON.                                                     UDT        355         I
          -FCC1088  -CCG1
UDT                 PL     X6,CMR1     IF IH[R2] \ I.GL  */ REF TO AP/IO LIST         UDT        356         I
          -CCG1
UDT                 BX6    X7-X2                                                      UDT        357         I

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN UDT

```
          -FCC1088  -CCG1
UDT                 SX6    X6                                                UDT          358          I
          -FCC1088  -CCG1
UDT                 NZ     X6,CMR2     IH[R2] " CON.                         UDT          359          I
          -FCC1088  -CCG1
UDT                                                                           UDT          360          I
          -FCC1088  -CCG1
UDT                 SX7    X7+         IHF = IH[R2]  */ REMOVE *CA*          UDT          361          I
          -FCC1088  -CCG1
UDT                 LX6    B1,X7                                             FCC1088       29          I
          -CCG1
UDT                 SB5    X6                                                FCC1088       30          I
          -CCG1
UDT                 SA2    A0-B5       WORDB = SYM(IH[R2])                   FCC1088       31          I
          -CCG1
UDT                 SB5    B1          CL = 1                                FCC1088       32          I
          -CCG1
UDT                 LX2    59-P.LDO                                          FCC1088       33          I
          -CCG1
UDT                 SX6    X7                                                CCG1         397     A
UDT                 AX6    IH.IP                                             CCG1         398     A
UDT                 NZ     X6,CMR1     IF I[IH] " 0  */ REF TO GL/AP LIST    CCG1         399     A
UDT                                                                          CCG28         45     A
UDT                 NZ     B3,CMR6     IF C0 " 0     */ INDEXED REF          CCG28         46     A
UDT                 SB5    B4+X7                                             CCG1         400     A
UDT                 LX6    B1,X7                                             CCG1         401     A
UDT                 SA2    B5+X6       WORDB = SYM(3*IH[R2W]+1)              CCG1         402     A
UDT                 SB5    B1          CL = 1                                CCG1         403     A
UDT                 LX2    59-WB.LDOP                                        CCG1         404     A
UDT                 MI     X2,CMR5     IF LDO[WORDB] */ VAR IS LOAD ONLY     FCC1088       34     A
UDT                 LX2    WB.LDOP-WB.FPP                                    CCG23        143     A
UDT                 PL     X2,CMR2     IF ^FP[DI]                            CCG23        144     A
UDT                 SA2    A2+B1       WORDC = WORDB + 1                     CCG23        145     A
UDT                 SX3    B1                                                CCG23        146     A
UDT                 IX6    X2+X3       RA[WORDC] = RA[WORDC] + 1  */ COUNT REFS  CCG23    147     A
UDT                 SA6    A2                                                CCG23        148     A
UDT                                                                          UDT          362     A
UDT       CMR2      SA2    O.UDB+X4    CI = UDB(HI)                          UDT          363     A
UDT                 ZR     X2,CMR4     IF CI = 0     */ BASE TABLE ENTRY IS EMPTY  UDT    364     A
UDT                                                                          UDT          365     A
UDT       *         SEARCH HASH CHAIN                                        UDT          366     A
UDT                                                                          UDT          367     A
UDT                 SA7    B7          [UDI] = IHF   */ SEARCH TERMINATOR    UDT          368     A
UDT                 SB3    X2          C0 = CI       */ INDEX TO LAST ON CHAIN  UDT       369     A
UDT                                                                          UDT          370     A
UDT       CMR3      SA3    B7+X2       UDW = UDI(CI)                         UDT          371     A
UDT                 SB5    X2          CL = CI                               UDT          372     A
UDT                 BX6    X3-X7                                             UDT          373     A
UDT                 LX3    -UA.HLP                                           UDT          374     A
UDT                 SX2    X3          CI = HL[UDW]                          UDT          375     A
UDT                 BX4    -X1*X6                                            UDT          376     A
UDT                 NZ     X4,CMR3     IF CAIH[UDW] " IHF                    UDT          377     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN UDT

```
UDT                                                                                  UDT        378    A
UDT                NZ     B5,CMR5     IF CL " 0     */ ENTRY IN TABLE                 UDT        379    A
UDT                                                                                  UDT        380    A
UDT       *        ADD  *CAIH* TO *UDI*                                              UDT        381    A
UDT                                                                                  UDT        382    A
UDT        CMR4    SX4    B3                                                         UDT        383    A
UDT                SX6    B6          UL = UI                                        UDT        384    A
UDT                LX4    UA.HLP                                                     UDT        385    A
UDT                BX7    X4+X7                                                      UDT        386    A
UDT                SA7    B7+B6       UDI(UI) = UDIW(0,C0,IHF)                       UDT        387    A
UDT                SB6    B6+B1       UI = UI + 1                                    UDT        388    A
UDT                SB5    X6          CL = UL                                        UDT        389    A
UDT                SA6    A2          UDB(HI) = UL  */ UPDATE BASE TABLE             UDT        390    A
UDT                                                                                  UDT        391    A
UDT       *        CHAIN ENTRY TO *UDI*                                              UDT        392    A
UDT                                                                                  UDT        393    A
UDT        CMR5    SX4    B5                                                         UDT        394    A
UDT                LX4    R1.INP+1                                                   UDT        395    A
UDT                BX6    X4+X5       IN[R1] = 2*CL                                  UDT        396    A
UDT                SA6    A5                                                         UDT        397    A
UDT                EQ     CMR1                                                       UDT        398    A
UDT                                                                                  UDT        399    A
UDT       *        INDEXED MEMORY REF                                                UDT        400    A
UDT                                                                                  UDT        401    A
UDT        CMR6    SB5    B1          CL = 1                                         UDT        402    A
UDT                EQ     B3,B5,CMR5  IF RF[R2] = 1                                  UDT        403    A
UDT                SA3    CMRA                                                       UDT        404    A
UDT                SB3    B0          C0 = 0                                         UDT        405    A
UDT                BX7    X3+X7       CR[IHF] = 1;  CA[IHF] = -0                     UDT        406    A
UDT                EQ     CMR2                                                       UDT        407    A
UDT                                                                                  UDT        408    A
UDT        CMRA    BFMW   UA,(CR,CA)                                                 UDT        409    A
UDT        CPL     TITLE  CPL - CHAIN PARAMETER LIST ENTRIES TO *UDI*                UDT        410         I
          -CCG1
UDT       **       CPL - CHAIN PARAMETER LIST ENTRIES TO *UDI*                       UDT        411         I
          -CCG1
UDT        CG.CPL  TITLE  CG.CPL - CHAIN PARAMETER LIST ENTRIES TO *UDI*             CCG1       405         I
          -CCGNC
UDT        CG$CPL  TITLE  CG$CPL - CHAIN PARAMETER LIST ENTRIES TO *UDI*             CCGNC      863    A
UDT       **       CG.CPL - CHAIN PARAMETER LIST ENTRIES TO *UDI*                    CCG1       406         I
          -CCGNC
UDT       **       CG$CPL - CHAIN PARAMETER LIST ENTRIES TO *UDI*                    CCGNC      864    A
UDT       *                                                                          UDT        412    A
UDT       *        ENTRY  (X5) = INDEX , TABLE INDEX OF PARAMETER LIST               UDT        413    A
UDT       *               (X6) = TBL , ADDRESS OF TABLE POINTER                      UDT        414    A
UDT       *               (B2) = LEN , NUMBER OF ENTRIES IN TABLE                    UDT        415    A
UDT       *                                                                          UDT        416    A
UDT       *        EXIT   (X1) = INDX , TO LIST IN *IOL*                             UDT        417    A
UDT       *               (X0) = LEN , OF LIST IN *IOL*                              UDT        418    A
UDT       *               (X7) = L.IOL                                               UDT        419    A
UDT       *               (B5) = *UDI* INDEX OF LAST ENTRY IN ARG LIST               UDT        420    A
UDT                                                                                  UDT        421    A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN UDT

```
UDT        CPL      ENTRY.                                                    UDT          422          I
          -CCG1
UDT        CG.CPL   ENTRY.                                                    CCG1         407          I
          -CCGNC
UDT        CG$CPL   ENTRY.                                                    CCGNC        865     A
UDT                 LX5    18                                                 UDT          423     A
UDT                 PX4    B2,X5                                              UDT          424     A
UDT                 SX1    B2+B2                                              UDT          425     A
UDT                 BX5    X4+X6                                              UDT          426     A
UDT                 ALLOC  IOL,X1      ALLOC( IOL , 2*LEN )                   UDT          427     A
UDT                 SX0    B6                                                 UDT          428     A
UDT                 ALLOC  UDI,X1      ALLOC( UDI , 2*LEN );  UI = OLD(L.UDI) UDT          429     A
UDT                 SB4    X2          UDI = O.UDI                            UDT          430     A
UDT                 UX7    B2,X5       N = LEN                                UDT          431     A
UDT                 SA2    O.IOL                                              UDT          432     A
UDT                 IX1    X2+X0       S0 = O.IOL + OLD(L.IOL)                UDT          433     A
UDT                 SA0    X1          SI = S0       */ STORE INDEX           UDT          434     A
UDT                 SA4    X5                                                 UDT          435     A
UDT                 AX7    18                                                 UDT          436     A
UDT                 IX6    X4+X7                                              UDT          437     A
UDT                 MX0    -IH.CAIHL                                          UDT          438     A
UDT                 SA5    X6          TI = O.TBL + INDEX                     UDT          439          I
          -FCC1088
UDT                 SA3    O.SYM                                              FCC1088       35     A
UDT                 SA5    X6+         TI = O.TBL + INDEX                     FCC1088       36     A
UDT                 SB7    X3-1                     (B7) = SYM-1              FCC1088       37          I
          -CCG1
UDT                 SB7    X3+1                     (B7) = SYM + 1            CCG1         408     A
UDT                                                                           UDT          440     A
UDT        CPL1     BX7    -X0*X5      IHF = CAIH[TI]                         UDT          441     A
UDT                 LX5    59-AP.CRP                                          UDT          442     A
UDT                 PL     X5,CPL2     IF ^CR[TI]                             UDT          443     A
UDT                 SA3    CMRA                                               UDT          444     A
UDT                 LX5    1+AP.CRP                                           UDT          445     A
UDT                 BX7    X3+X7       CR[IHF] = 1;  CA[IHF] = -0             UDT          446     A
UDT                 EQ     CPL3                                               UDT          447     A
UDT                                                                           UDT          448     A
UDT        CPL2     LX5    AP.CRP-AP.P1P                                      UDT          449     A
UDT                 MI     X5,CPL4     IF P1[TI]     */ DOUBLE / COMPLEX      UDT          450     A
UDT                 LX5    1+AP.P1P                                           UDT          451     A
UDT                                                                           UDT          452     A
UDT        CPL3     RJ     SUI                                                UDT          453     A
UDT                 BX2    X2*X5                                              UDT          454     A
UDT                 SB2    B2-B1       N = N - 1                              UDT          455     A
UDT                 BX6    X2+X6                                              UDT          456     A
UDT                 SA5    A5+B1       TI = TI + 1                            UDT          457     A
UDT                 SA6    A0          [SI] = IO[TI] ! SUI( IHF )             UDT          458          I
          -FCC0789
UDT                 SA6    A0          [SI] = (IO,USE)[TI] ! SUI( IHF )       FCC0789       28     A
UDT                 SA0    A0+B1       SI = SI + 1                            UDT          459     A
UDT                 NZ     B2,CPL1     IF N " 0                               UDT          460     A
UDT                                                                           UDT          461     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN UDT

```
UDT       *         SET EXIT CONDITIONS                                          UDT        462    A
UDT                                                                              UDT        463    A
UDT                 SA2    O.IOL                                                 UDT        464    A
UDT                 SB5    X6                                                    UDT        465    A
UDT                 SX3    A0                                                    UDT        466    A
UDT                 SX6    B6                                                    UDT        467    A
UDT                 SA6    L.UDI       L.UDI = UI                                UDT        468    A
UDT                 IX7    X3-X2       L.IOL = SI - O.IOL                        UDT        469    A
UDT                 IX0    X3-X1       LEN = SI - S0 */ LIST LENGTH              UDT        470    A
UDT                 SA7    L.IOL                                                 UDT        471    A
UDT                 IX1    X1-X2       INDX = S0 - O.IOL                         UDT        472    A
UDT                 EQ     CPL                                                   UDT        473         I
          -CCG1
UDT                 EQ     CG.CPL                                                CCG1       409         I
          -CCGNC
UDT                 EQ     CG$CPL                                                CCGNC      866    A
UDT                                                                              UDT        474    A
UDT       *         DOUBLE/COMPLEX ENTRY                                          UDT        475    A
UDT                                                                              UDT        476    A
UDT        CPL4     LX5    1+AP.P1P                                              UDT        477    A
UDT                 RJ     SUI                                                   UDT        478    A
UDT                 BX2    X2*X5                                                 UDT        479    A
UDT                 SX4    B1                                                    UDT        480    A
UDT                 BX6    X2+X6                                                 UDT        481    A
UDT                 LX4    IH.CAP                                                UDT        482    A
UDT                 SA6    A0          [SI] = IO[TI] < SUI( IHF )                UDT        483         I
          -FCC0789
UDT                 SA6    A0          [SI] = (IO,USE)[TI] ! SUI( IHF )          FCC0789     29    A
UDT                 SA0    A0+B1       SI = SI + 1                               UDT        484    A
UDT                 IX7    X4+X7       CA[IHF] = CA[IHF] + 1 */ CA ALWAYS POSITIVE UDT      485    A
UDT                 EQ     CPL3                                                  UDT        486    A
UDT        SUI      TITLE  SUI - SET *UDI* INDEX                                 UDT        487    A
UDT       **        SUI - SET *UDI* INDEX                                        UDT        488    A
UDT       *                                                                      UDT        489    A
UDT       *         ENTRY  (X0) = MASK(-UA.CAIHL)                                UDT        490    A
UDT       *                (X7) = IHF , CA,IH OF LOCATION                        UDT        491    A
UDT       *                (B4) = O.UDI                                          UDT        492    A
UDT       *                (B6) = UI , NEXT AVAIL LOC IN *UDI*                   UDT        493    A
UDT       *                (B7) = O.SYM - 1                                      FCC1088     38    A
UDT       *                                                                      UDT        494    A
UDT       *         EXIT   (X6) = U0 = INDEX TO IHF IN *UDI*                     UDT        495    A
UDT       *                (X2) = 1S59                                           UDT        496         I
          -FCC0789
UDT       *                (X2) = MASK(60-AP.USEP)  ( 3S58 )                     FCC0789     30    A
UDT       *                                                                      UDT        497    A
UDT       *         USES   B - 3, 5                                              UDT        498    A
UDT       *                A - 2, 3, 6                                           UDT        499    A
UDT       *                X - 2, 3, 4, 6                                        UDT        500    A
UDT                                                                              UDT        501    A
UDT        SUI0     SX6    B5+B5       U0 = 2*CL                                 UDT        502    A
UDT                 MX2    1                                                     UDT        503         I
          -FCC0789

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN UDT

```
UDT                MX2    -AP.USEP                                          FCC0789      31    A
UDT                                                                         UDT         504    A
UDT       SUI      ROUTINE                                                  UDT         505    A
UDT                LX6    B1,X7                                             FCC1088      39    A
UDT                SB3    X6                                                FCC1088      40          I
          -CCG1
UDT                SA2    B7-B3       WORDB = SYM(IH)                       FCC1088      41          I
          -CCG1
UDT                SB5    B1          CL = 1        */ SET FOR LD ONLY CLASS FCC1088      42          I
          -CCG1
UDT                LX2    59-P.LDO                                          FCC1088      43          I
          -CCG1
UDT                SB3    B7+X6                                             CCG1        410    A
UDT                SA2    B3+X7       WORDB = SYM(3*IH+1)                   CCG1        411    A
UDT                SB5    B1          CL = 1                                CCG1        412    A
UDT                LX2    59-WB.LDOP                                        CCG1        413    A
UDT                MI     X2,SUI0     IF LDO[WORDB] */ VAR IS LD ONLY       FCC1088      44    A
UDT                MX4    -6                                                UDT         506    A
UDT                BX6    -X4*X7      HI = IHF & 77B                        UDT         507    A
UDT                SA7    B4          [UDI] = IHF   */ SET SEARCH TERMINATOR UDT        508    A
UDT                SB3    B0          C0 = 0                                UDT         509          I
          -FCC1088
UDT                SA2    O.UDB+X6    CI = UDB(HI)                          UDT         510    A
UDT                SX4    0                                                 UDT         511          I
          -FCC1088
UDT                SX4    B0          C0 = 0                                FCC1088      45    A
UDT                ZR     X2,SUI2     IF CI = 0     */ BASE TABLE ENTRY IS EMPTY UDT    512    A
UDT                SB3    X2+         C0 = CI                               UDT         513          I
          -FCC1088
UDT                SB3    X2          C0 = CI                               FCC1088      46    A
UDT                                                                         UDT         514    A
UDT       *        SEARCH HASH CHAIN                                        UDT         515    A
UDT                                                                         UDT         516    A
UDT       SUI1     SA3    B4+X2       UDW = UDI(CI)                         UDT         517    A
UDT                SB5    X2          CL = CI                               UDT         518    A
UDT                BX6    X3-X7                                             UDT         519    A
UDT                LX3    -UA.HLP                                           UDT         520    A
UDT                SX2    X3          CI = HL[UDW]                          UDT         521    A
UDT                BX4    -X0*X6                                            UDT         522    A
UDT                NZ     X4,SUI1     IF CAIH[UDW] " IHF                    UDT         523    A
UDT                                                                         UDT         524    A
UDT                NZ     B5,SUI0     IF CL " 0     */ ENTRY IN TABLE       UDT         525    A
UDT                                                                         UDT         526    A
UDT       *        ADD ENTRY TO *UDI*                                       UDT         527    A
UDT                                                                         UDT         528    A
UDT                SX4    B3                                                UDT         529    A
UDT                LX4    UA.HLP                                            UDT         530    A
UDT       SUI2     BX6    X4+X7       UDI(UI) = UDIW(0,C0,IHF)              UDT         531    A
UDT                SA6    B4+B6                                             UDT         532    A
UDT                SX6    B6          U0 = UI                               UDT         533    A
UDT                MX2    1                                                 UDT         534          I
          -FCC0789

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN UDT

```
UDT                MX2    -AP.USEP                                                  FCC0789      32     A
UDT                SB6    B6+B1       UI = UI + 1                                   UDT         535     A
UDT                SA6    A2          UDB(HI) = U0                                  UDT         536     A
UDT                LX6    1           U0 = 2*U0                                     UDT         537     A
UDT                EQ     SUI                                                       UDT         538     A
UDT       PBB      TITLE  PBB - PROCESS BASIC BLOCK FOR OPT = 2                     UDT         539     A
UDT      **        PBB - PROCESS BASIC BLOCK FOR OPT = 2                            UDT         540     A
UDT      *                                                                          UDT         541     A
UDT      *         SETUP *BIT* ENTRY FOR *TXT* IN BLOCK AND DUMP BLOCK TO MASS      UDT         542     A
UDT      *         STORAGE.                                                         UDT         543     A
UDT      *                                                                          UDT         544     A
UDT      *         ENTRY  *TXT* - SQUEEZED BASIC BLOCK                              UDT         545     A
UDT      *                *IOL* - PARAMETER LIST INFO                               UDT         546     A
UDT      *                                                                          UDT         547     A
UDT      *         EXIT   BLOCK WRITTEN TO MASS STORAGE                             UDT         548     A
UDT      *                R1[*BOS*] = *BIT* INDEX OF BLOCK                          UDT         549     A
UDT      *                R2[*BOS*] = 42/0,18/L.TXT, INSTRUCTION LENGTH             UDT         550     A
UDT                                                                                 UDT         551     A
UDT       PBB      ENTRY.                                                           UDT         552           I
          -CCGNC
UDT       PBB      ENTRY. **,#                                                      CCGNC       867     A
UDT                CALL   CMR         CHAIN MEMORY REFS TO *UDI*                    UDT         553     A
UDT                                                                                 UDT         554     A
UDT      *         SET INFO IN R1, R2 WORDS OF *BOS*                                UDT         555     A
UDT                                                                                 UDT         556     A
UDT                SA3    O.TXT                                                     UDT         557     A
UDT                SA2    L.TXT                                                     UDT         558     A
UDT                SA1    L.IOL                                                     UDT         559     A
UDT                SA5    X3          R1 = [O.TXT]                                  UDT         560     A
UDT                MX0    -IH.IHL                                                   UDT         561     A
UDT                SA4    L.BIT                                                     UDT         562     A
UDT                BX5    X0*X5                                                     UDT         563     A
UDT                IX7    X4+X5       IH[R1] = L.BIT  */ SAVE BLOCK NUMBER          UDT         564     A
UDT                SA7    A5                                                        UDT         565     A
UDT                BX6    X2                                                        UDT         566     A
UDT                LX6    R2.TXTLP                                                  UDT         567     A
UDT                SA6    A7+B1       R2 = R1 + 1;  TXTL[R2] = L.TXT                UDT         568     A
UDT                                                                                 UDT         569     A
UDT                IX7    X1+X2                                                     UDT         570     A
UDT                SA3    =XM.BBL                                                   UDT         571           I
          -CCGNC
UDT                SA3    =XM.BBL#                                                  CCGNC       868     A
UDT                IX0    X3-X7                                                     UDT         572     A
UDT                PL     X0,PBB0     M.BBL = MAX( M.BBL , L.TXT + L.IOL )          UDT         573     A
UDT                SA7    A3+                                                       UDT         574     A
UDT                                                                                 UDT         575     A
UDT       PBB0     ZR     X1,PBB1     IF L.IOL = 0  */ NO PARAMETER LISTS           UDT         576     A
UDT                ALLOC  TXT,X1                                                    UDT         577     A
UDT                SX3    X2+B6                                                     UDT         578     A
UDT                SA2    O.IOL                                                     UDT         579     A
UDT                MOVE   X1,X2,X3    MOVE( L.IOL , O.IOL , O.TXT + OLD(L.TXT) )    UDT         580     A
UDT                MX6    0                                                         UDT         581     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN UDT

```
UDT                SA6    L.IOL       L.IOL = 0                                    UDT          582     A
UDT                SA6    =XPIOL      PIOL = 0                                     UDT          583          I
          -CCG1
UDT                SA6    =XCO.PIOL    PIOL = 0                                    CCG1         414          I
          -CCG7
UDT                                                                               UDT          584     A
UDT       *        SETUP *BIT* ENTRY, AND DUMP BLOCK TO MASS STORAGE              UDT          585     A
UDT                                                                               UDT          586     A
UDT        PBB1    ALLOC  BIT,1       ALLOC( BIT , 1 )                            UDT          587     A
UDT                SA4    O.TXT       FSCM = [O.TXT]  */ FWA OF BLOCK             UDT          588     A
UDT                SB5    X2+B6       BI = O.BIT + OLD(L.BIT)                     UDT          589     A
UDT                SA3    X4          R1 = [O.TXT]                                UDT          590     A
UDT                LX3    -R1.INP                                                 UDT          591     A
UDT                SX6    X3                                                      UDT          592     A
UDT                LX6    BI.FLNP     FLN[BI] = IN[R1]                            UDT          593     A
UDT                SA1    X4+4        R1 = R1 + 4   */ LOOKAT FIRST INSTRUCTION   UDT          594     A
UDT                UX7    B2,X1                                                   UDT          595     A
UDT                SB3    B2-OC.LAB                                               UDT          596     A
UDT                NZ     B3,PBB2     IF OC[R1] " OC.LAB                          UDT          597     A
UDT                SX0    B1                                                      UDT          598     A
UDT                SB4    X1          LIH = IH[R1]                                UDT          599     A
UDT                LX0    BI.ILP      IL[BI] = 1                                  UDT          600     A
UDT                BX6    X0+X6                                                   UDT          601     A
UDT                                                                               UDT          602     A
UDT        PBB2    SA2    X4+B1       LTXT = [O.TXT+1]                            UDT          603     A
UDT                SB7    X2-8                                                    UDT          604     A
UDT                SA5    X4+B7       R1L = O.TXT + LTXT-8                        UDT          605     A
UDT                UX7    B2,X5                                                   UDT          606     A
UDT                SX0    B2-OC.UJP                                               UDT          607     A
UDT                NZ     X0,PBB3     IF OC[R1L] " OC.UJP                         UDT          608          I
          -FCC1755X
UDT                SX0    B1                                                      UDT          609          I
          -FCC1755X
UDT                ZR     X0,PBB2A    IF OC[RI] = OC.UJP                          FCC1755X       5     A
UDT                SX0    B2-OC.JIN                                               FCC1755X       6     A
UDT                NZ     X0,PBB3      IF OC[R1L] " OC.JIN                        FCC1755X       7     A
UDT        PBB2A   SX0    B1                                                      FCC1755X       8     A
UDT                LX0    BI.FJP                                                  UDT          610     A
UDT                BX6    X0+X6       FJ[BI] = 1    */ INDICATE FINAL *UJP*       UDT          611     A
UDT                SX7    X2-12                                                   UDT          612     A
UDT                NZ     X7,PBB3     IF LTXT " 12  */ NOT 2 INSTRUCTION BLOCK    UDT          613     A
UDT                NZ     B3,PBB3     IF OC[R1] " OC.LAB                          UDT          614     A
UDT                LX0    BI.LJPP-BI.FJP                                          UDT          615     A
UDT                BX6    X0+X6       LJP[BI] = 1   */ INDICATE LAB/JP ONLY       UDT          616     A
UDT                                                                               UDT          617     A
UDT        PBB3    SA5    L.TXT                                                   UDT          618     A
UDT                SA3    =XL.PTXT                                                UDT          619          I
          -CCGNC
UDT                SA3    =XL.PTXT#                                               CCGNC        869     A
UDT                IX7    X3+X5       L.PTXT = L.PTXT + L.TXT                     UDT          620     A
UDT                SA7    A3                                                      UDT          621     A
UDT                                                                               UDT          622     A

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

```
          LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN UDT

UDT       *         INHIBIT COMBINATION OF LARGE BLOCKS WITH EXTERNAL REFS IN CBB          UDT          623     A
UDT                                                                                        UDT          624     A
UDT                 SX7    X5-2004B                                                        UDT          625     A
UDT                 SA3    =XXR                                                            UDT          626          I
          -CCG7
UDT                 SA3    =XCG.XR                                                         CCG7         140          I
          -CCGNC
UDT                 SA3    =XCC$XR                                                         CCGNC        870     A
UDT                 MI     X7,PBB4     IF L.TXT < 2000B                                    UDT          627     A
UDT                 LX3    1+BI.FJP                                                        UDT          628     A
UDT                 BX6    X3+X6       FJ[BI] = XR                                         UDT          629     A
UDT                                                                                        UDT          630     A
UDT       PBB4      SX7    X5+2        L = L.TXT + 2                                       UDT          631     A
UDT                 IX0    X4+X5       LTXT = O.TXT + L.TXT                                UDT          632     A
UDT                 SA6    B5                                                              UDT          633     A
UDT                 LX7    BT.LENP                                                         UDT          634     A
UDT                 SA7    X0          [LTXT] = BTW(0,0,0,L)  */ TRAILER WORD              UDT          635     A
UDT                 SX3    B6                                                              UDT          636     A
UDT                 LX3    BH.BIP                                                          UDT          637     A
UDT                 LX7    BH.LENP-BT.LENP                                                 UDT          638     A
UDT                 BX7    X3+X7                                                           UDT          639     A
UDT                 SA7    X4-1        [O.TXT-1] = BHW(0,OLD(L.BIT),0,L)                   UDT          640     A
UDT                 SB2    0           FLCM = 0                                            UDT          641     A
UDT                 CALL   SMB         SMB( O.TXT , L.TXT )                                UDT          642          I
          -CCGNC
UDT                 CALL   SMB#        SMB( O.TXT , L. TXT )                               CCGNC        871     A
UDT                 ADDWRD BIT,X1      ADDWRD( BIT , RIW )                                 UDT          643     A
UDT                 EQ     PBB                                                             UDT          644     A
UDT                                                                                        UDT          645     A
UDT                 ENTRY  F.UDT                                                           UDT          646          I
          -CCGNC
UDT                                                                                        UDT          647          I
          -CCGNC
UDT        F.UDT    END                                                                    UDT          648          I
          -CCGNC
UDT                 ENTRY  F$UDT                                                           CCGNC        872     A
UDT                                                                                        CCGNC        873     A
UDT        F$UDT    END                                                                    CCGNC        874     A

     SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - UDT

          IDENTIFIER                TOTAL    ACTIVE

          UDT                         648       576
          FCC0789                       5         5
          FCC0807                       6         1
          FCC1088                      20         9
          FCCT005                       6         5
          CCG1                         39        28
          CCG4                         91        53
          CCG7                          6         0
          CCG8                         12         5

          0         1         2         3         4         5         6         7         8
          12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

LIST OF CONTROL, ACTIVE, AND/OR INACTIVE CARDS IN UDT

CCG9	2	0
CCGNC	39	37
CCG11	7	6
FCC1755X	4	4
CCG16	3	3
CCG18	3	2
CCG20	1	1
CCG21	8	7
CCG23	6	6
CCG27	2	2
CCG28	2	2
CCG31	1	1
CCG35	2	2
CCGA011	9	1
CCGA012	43	43
CCGA025	11	11
CCGA026	2	2
F1037CC	5	5

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

SUMMARY OF UPDATE IDENTIFIERS WITHIN DECK - TOTALS

IDENTIFIER	TOTAL	ACTIVE
YANK$$$	0	0
COMDKS	1	1
CCOMISC	54	53
CCOMLOL	171	170
CGHCRMD	148	148
COMADEF	210	210
COMAREG	228	227
FA=CLO	38	38
FA=DEFS	401	401
FA=EOF	45	45
FA=EOR	45	45
FA=FLSH	60	60
FA=OPE	61	60
FA=RDC	76	71
FA=RDW	126	125
FA=RWX	53	53
FA=SET	194	194
FA=WTW	111	111
OPRDEFS	165	87
PSODEFS	53	53
SSTCALL	5	5
TEXTS	1	1
CCG	1	1
CGIA	5021	4232
MACROS	570	567
CGTM	1013	810
CT	1	1
CFTBLD	71	60
IOTBLD	55	55

GPOCOM	182	176
UDTBLD	54	51
MIO	2079	1862
FBV	311	294
GPO	6191	5401
GRA	6290	5629
PROSEQ	2631	1752
SQZ	2866	2384
MCG	4535	4145
BDT	1671	1511
DMPIIT	102	102
DMPRLST	68	67
DMPSIT	53	53
DMPTREE	54	54
DMPUDI	70	69
PRNTGRA	202	200
PRNTMIO	286	286
PRNTUDI	47	47
PRNTABV	49	49
PRNTBV	83	83
PRNTRLI	175	173
PRNTMCG	264	264
CFA	1535	1402
UDT	648	576
FCC0391	4	4
FCC0392	144	140
FCC0393	34	34
FCC0408	101	65
FCC0414	77	74
FCC0415	43	31
FCC0448	184	139
FCC0502	15	15
FCC0504	1	0
FCC0508	16	16
FCC0509	61	58
FCC0513	31	28
FCC0534	5	3
FEAT121A	2	0
FCC0514	751	610
FCC0535	9	9
FCC0584	19	19
FCC0586	18	18
FCC0588	9	9
FCC0594	20	15
FCC0603	117	113
FCC0612	2	2
FCC0613	21	21
FCC0625	18	12
FCC0634	8	7
FCC0640	5	5
FCC0641	88	88
FCC0642	8	8
FCC0643	74	70
FCC0645	32	30
FCC0646	62	59
FCC0651	12	12
FCC0658	5	0
FCC0602	17	17

FCC0656	48	41
FCC0716	43	43
FCC0672	4	2
FCC0675	238	222
FCC0715	7	7
FCC0719	1	1
FCC0735	22	22
FCC0789	12	12
FCC0791	24	5
FCC0792	22	22
FCC0717	258	242
FCC0764	15	15
FCC0811	40	40
FCC0812	10	10
FCC0814	3	3
FCC0839	3	3
FCC0841	6	6
FCC0872	3	3
FCC0882	7	6
FCC0883	2	2
FCC0885	6	3
FCC0912	36	34
FCC0919	4	4
S3143FC	7	0
FCC0715A	8	8
FCC0777	6	0
FCC0881	250	231
FCC0909	73	71
FCC0914	88	88
FCC0977	93	91
FCC1014	100	97
FCC1025	34	34
FEAT155A	21	18
FEAT155AB	4	1
FCC0957	14	14
FCC0985	60	50
FEAT155F	10	4
S3143FCA	1	0
FCC0807	6	1
FCC1025A	2	2
FCC1029	6	6
FCC1039	28	28
FCC1040	6	6
FCC1056	57	50
FCC1085	22	22
FCC1088	24	13
FCC1107	7	7
FCC1123	4	4
FCC1126	108	54
FCC1127	24	24
FCC1191	27	27
FCC1196	6	6
FCC1227	9	8
FCC1239	5	5
FCC1254	126	125
FCC1290	24	22
FCCT001	2	0
FCC1319	60	40

FCC1322	6	2
FCC1352	2	1
FCC1353	18	17
FCC1413	11	11
FCC1424	2	1
FCC1442	9	9
FCC1451	13	10
F7540FT	35	30
FCC1491X	4	2
FCC1498X	2	2
FCC1530X	12	12
RSM2763X	304	104
FCCT005	6	5
FCC1504X	25	24
FCC1531X	3	3
FCC1532X	7	7
FCC1555X	86	83
FCC1556X	13	7
FCC1518X	32	30
RSM2763AX	5	0
FCC1568X	3	2
FCC1580X	5	5
CCG1	356	156
CCG2	46	45
CCG3	190	188
CCG4	107	57
CCG5	67	30
MCGF1	11	9
FCC1596X	3	3
FCC1623X	3	0
CCGDBG	2	1
FCC1626	5	5
FCC1644X	16	16
FCC1646X	21	21
SMOCDEF	22	22
CCG7	61	20
CCG8	39	25
CCG9	10	7
HISTORY	2	2
CCOMGCM	125	93
CCGDBGM	195	195
CCGILFD	219	217
FCC0885XA	15	15
FCC1655X	26	26
FCC1656X	40	40
FCC1660X	31	31
FCC1584X	15	15
FCC1677X	9	9
FCC1695X	10	10
FCC1700X	3	3
FCC1710X	27	27
FCC1719X	9	9
CCGNC	395	373
CCG11	34	32
CCG12	1	1
FCC1739X	6	6
FCC1740X	16	15
CMPLTXT	74	65

CCGTEXT	600	584
CCG13	6	6
CCG14	13	12
FCC1755X	8	8
FCC1770X	17	17
FCC1825X	28	28
FCC1828X	9	9
FCC1837X	31	31
FCC1842X	10	0
FCC1845X	7	7
FCC1858X	4	4
CCGTMTV	90	87
CCG15	721	654
CCG16	293	275
CCG17	925	886
FCC0238AX	52	52
CCG18	185	177
CCG19	153	145
CCG*78	7	5
FCC1908	76	76
FCC1918	27	27
FCC1928	11	11
CCGA002	8	8
CCGA005	47	45
CCG20	33	32
CCG21	81	62
CCG22	84	78
CCG23	148	144
CCGA006	39	39
CCGA007	27	27
CCG24	26	26
FCC1997	15	15
CCG25	186	183
CCG26	79	78
OUTPTK	1204	1202
CCG27	107	106
CCG28	46	41
FCC2038	5	5
FCC2045	13	12
FCC2056	15	15
FCC2051	18	17
CCG29	13	10
CCGA008	5	5
FCC2072	21	21
CCG30	6	6
CCG31	2	2
CCG33	0	0
COMAIDP	433	433
COMATOK	1552	1548
COMCBUB	471	471
COMCBUN	203	203
COMCIDP	8437	8435
COMCLFM	56	55
COMCMCS	78	78
COMCSBM	50	49
COMCTOK	1346	1346
COMDTOK	762	762
COMQSVR	436	436

COMSIDP	918	918
COMSTOK	197	197
CCOMRPV	373	372
DBG=MAC	412	410
DBG=IDP	6096	6096
CCG36	32	32
CCG32	10	10
CCG34	30	29
CCG35	71	46
FCC2074	12	12
FCC2091	5	5
FCCA462	10	10
CCG37	10	10
CCG38	16	16
CCG39	4	4
CCG40	24	24
FCC1040A	5	5
FCC2092	3	3
FCC2124	79	76
FCC2135	63	63
FCC2149	12	11
FCC2157	14	4
FCC2165	1	1
CCG41	35	32
L498	1	1
CCGA009	10	10
L505	1	1
F2950CC	14	14
CCGA010	10	10
CCGA011	120	112
CCGA012	52	52
CCGA013	5	5
CCGA014	5	5
CCGA015	5	5
L508	1	1
FCCA499X	55	55
FCCA506X	5	5
FCC2212X	6	6
CCGA017	13	13
CCGA018	118	117
CCGA019	144	137
CCGA021	65	65
CCGA022	5	5
CCGA024	21	21
L518	1	1
CCG*80	4	3
CCGA023	174	171
CCGA025	21	21
CCGA026	7	7
FCCA509X	40	40
FCCA513X	53	53
FCC2230X	13	13
FCC2271X	47	47
FCC2273X	49	49
L528	1	1
F1037CC	46	46
CCGA027	10	10
CCGA028	5	5

CCGA029	6	6
CCGA030	5	5
CCGA031	8	8
CCGA034	6	6
FCCA546X	10	10
FCCA550X	5	5
L538	1	1
CCG*81	4	3
CCGA035	16	16
CCGA036	18	18
CCGA037	7	7
CCGA039	27	27
L552	1	1
CCGA040	33	33
CCGA041	7	7
CCGA042	7	7
CCGA043	7	7
CCGA044	123	101
CCGA045	15	15
CCGA046	12	12
CCGA047	10	10
CCG0001	5	5
CCG0004	6	6
L564	1	1
CCG*82	4	3
CCGA049	22	22
CCGA052	18	18
CCGA054	9	9
CCG0006	12	12
CCG0013	15	15
L577	1	1
CCGA062	10	5
CCGA063	8	8
CCGA066	5	5
CCGA067	39	39
CCGA068	29	29
CCGA072	43	43
CCGA073	14	14
CCG0002	17	17
CCG0008	37	37
CCG0018	11	11
CCG0022	6	6
L587	1	1
CCGA076	6	6
FA=WTH	95	95
CCGA079	14	14
CCGA082	15	15
CCG0037	7	7
CCG0041	16	16
CCG0045	11	11
CCG0071	28	28
CCG0085	5	5
L601	1	1
CCGA083	579	570
CCG0012	14	14
CCG0034	28	28
CCG0057	14	14
CCG0086	12	12

CCG0088	25	25
CCG0089	15	15
L617	1	1
CCG0093	6	6
CCG0110	75	75
CCG0117	7	7
CCG0119	20	20
CCG0124	6	6
L628	1	1
CCGA085	10	10
CCGA088	7	7
CCG0127	51	51
CCG0128	20	20
CCG0146	6	6
CCG0147	7	7
CCG0150	9	9
CCG0154	12	12
L642	1	1
CCGA090	9	9
CCG0157	5	5
CCG0158	10	10
CCG0165	6	6
L650	1	1
CCG0172	18	18
CCG0181	7	5
L670	1	1
CCGA091	10	10
CCGA092	87	87
CCGA093	19	19
CCG0187	8	8
L688	1	1
CCGA094	9	9
CCG0177	13	13
CCG0188	6	6
CCG0189	7	7
CCG0191	32	32
CCG0193	13	13
CCG0196	6	6
L716	1	1
CCG0206	9	9
CCG0207	13	13
CCG0208	9	9
CCG0211	29	29
CCG0212	9	9
CCG0213	10	10
L739	1	1
CCG0214	7	7
CCG0215	10	10
CCG0216	18	18
CCG0218	19	19
CCG0220	10	10
L780	1	1
CCGA097	53	53
L797	1	1
CCGA099	29	14
CCG1*92	10	8
L803	1	1
CCGA100	22	22

L826	1	1
CCG*94	12	12
L840	1	1
L847	1	1
L851	1	1
L859	1	1
L871	1	1

11.25.08.UCLP, 01, 031, 106.368KLNS.

1412THE

1